VOT 71857

IMAGE ANALYSIS ON WIRE BONDING

(ANALISIS IMEJ UNTUK PENGIKATAN DAWAI)

NORLIZA MOHD NOOR

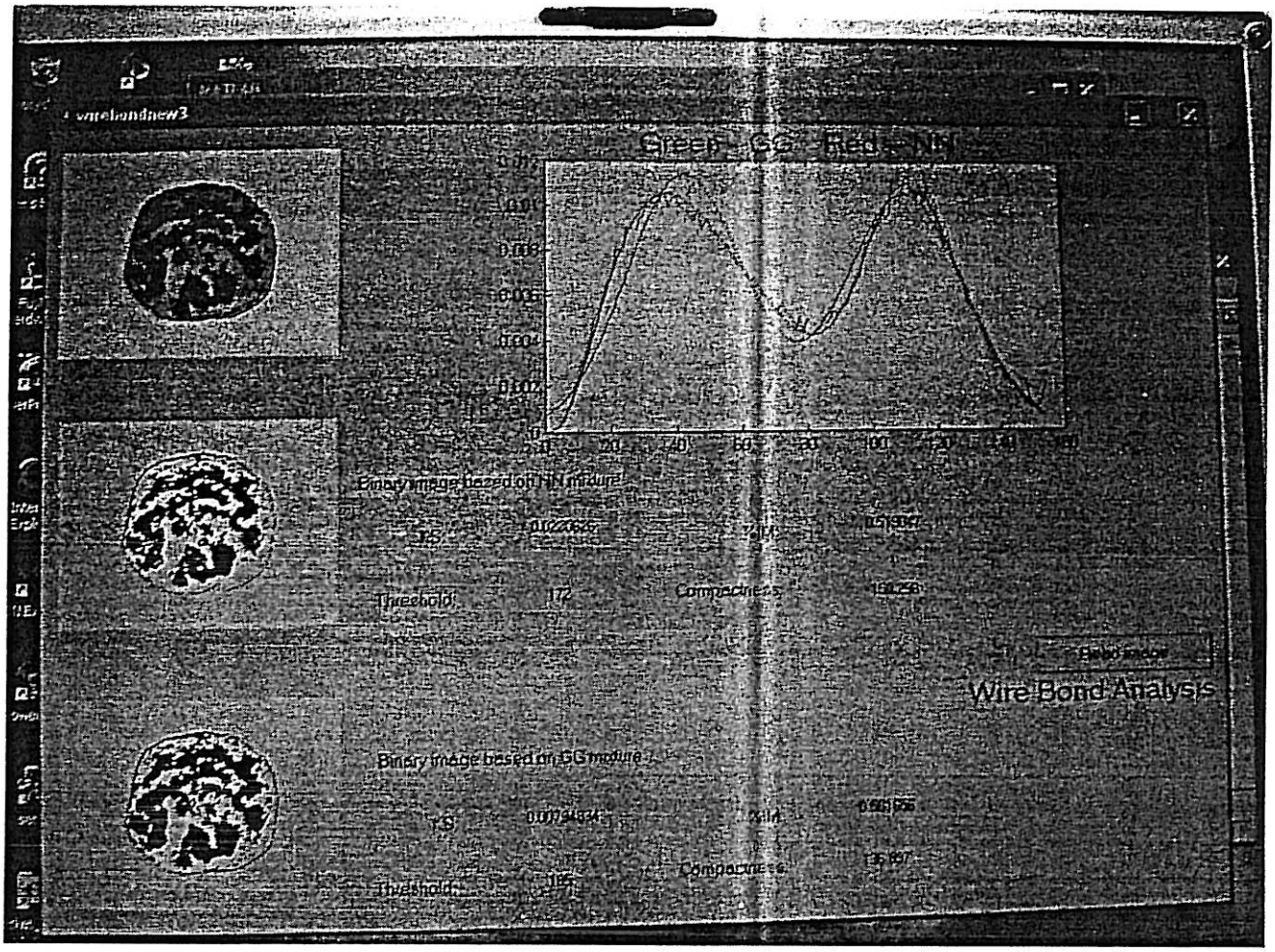
RESEARCH VOTE NO: 71857

DEPT. OF ELECTRICAL ENGINEERING
DIPLOMA PROGRAM STUDIES
UTM CITY CAMPUS
JALAN SEMARAK
54100 KUALA LUMPUR

2005

The project was carried with the collaboration from En. Omar Mohd. Badar, Texas Instrument (M) Sdn. Bhd., Assoc. Prof. Dr. Omar Mohd. Rijal from The Institute of Mathematical Science, Faculty of Science, University Malaya and his M.Sc. (by thesis and coursework) student, Mr. Liew Kian Wah. A copy of the thesis that was submitted to the Institute of Mathematical Science, Faculty of Science, University Malaya, is enclosed.

PRODUCT CREATED



A software was created to calculate the intermetallic area in the MATLAB software environment. The software able to give an exact percentage of the intermetallic area using 2 methods that is %IM – based on normal-normal mixture modeling and gamma-gamma mixture modeling. The software also gives a compactness number where a low value means good compact of the intermetallic area and high value means bad. Compactness will give user an indication on the quality of the wire bonding.

Modeling and Measuring Uniformity of Intermetalic Formation from the Au-Al Wire Bonding Image

Omar Mohd. Rijal*, Norliza Mohd. Noor†, Liew Kian Wah*, Omar Mohd. Badar§

* Institute of Mathematical Science, University of Malaya
†Diploma Program Studies, Universiti Teknologi Malaysia
§Texas Instruments (M) Sdn. Bhd.

The above titled has been submitted to the 5th WSEAS International Conference on Signal, Speech and Image Processing (SSIP 2005), that will be held in Corfu Island, Greece, August 17-19 2005.

Modeling and Measuring Uniformity of Intermetalic Formation from the Au-Al Wire Bonding Image

Omar Mohd. Rijal*, Norliza Mohd. Noor†, Liew Kian Wah*, Omar Mohd. Badar§
* Institute of Mathematical Science, University of Malaya
†Diploma Program Studies, Universiti Teknologi Malaysia
§Texas Instruments (M) Sdn. Bhd.
omarrija@um.edu.my, norliza@citycampus.utm.my, omar@ti.com

Abstract:- The quality of wire bonding is studied by investigating the relationship between intermetallic area and compactness. A wire-bonding image was converted into its grey-scaled equivalence. Two-component mixture distributions were fitted to the two-hump brightness histogram. The fitted models are used to find the optimum threshold points which was used in defining a binary image. Percentage intermetallic was calculated from this binary image. From the same image a statistic called compactness was defined as an indicator or measure of uniformity of intermetallic formation. The results suggest both percentage intermetallic and compactness should be considered while accessing the quality of the Au-Al bond.

Key-Words:- Wire Bonding, intermetallics formation, mixture distribution, percentage intermetallic, compactness.

1 Introduction

Wire bonding is an electrical interconnection technique in joining thin wires between electrical device and its conducting track. Gold (Au) and aluminium (Al) interconnections are commonly found in the process; in which thin gold wires are welded to an aluminium pad under a combination of heat, pressure and/or ultrasonic energy.

In examining the quality of wire bond, variables such as thickness [1], and circumference [2] of intermetallics have been studied. Suresh *et al.*[3] found that the bond quality is characterized by the intermetallic coverage in two ways that is, by the percentage of the intermetallic coverage and the uniformity of the coverage. Bonds with uniform intermetallic coverage show stronger bond shear strength than bonds with spotty intermetallic region. These latter 2 variables form our main focus in this study.

Colour wire bond images provided by a semiconductor production plant will be studied. These images were taken by high-powered microscope on the gold balls that were etched from the aluminium pad. Colours of the intermetallic compounds in the images are shown in Table 1, [4].

After some noise cleaning process and transformation, a brightness histogram can be drawn from the grey-scale image, where the horizontal axis represents the brightness levels while the vertical axis represents the number of pixels in the whole image with corresponding brightness level.

Table 1 Au-Al intermetallic compounds and their corresponding colours.

Intermetallic compound	Colour
Au_5Al_2	Tan
Au_4Al	Tan
Au_2Al	Tan
$AuAl$	White
$AuAl_2$	Purple

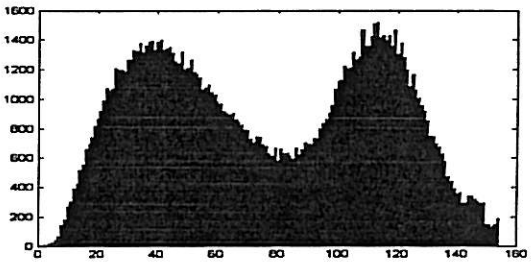


Fig. 1 Brightness histogram of a wire bond image.

2 The Normal-Normal Mixture Model

We fit a mixture distribution to the histogram with, $f(x)=\sum_{i=1}^{k} p_i g_i(x)$, where $0<p_i<1$ such that $\sum_{i=1}^{k} p_i = 1$ and $g_i(x)$ are probability density function of population π_i.

[5] has given a comprehensive description to the method in finding the maximum likelihood estimates of parameters. The maximum likelihood estimator for p_i is found to be

$$\hat{p}_i^{new} = \frac{1}{n}\sum_{j=1}^{n} \frac{\hat{p}_i^{old} g_i(x_j)}{f(x_j)} \qquad (1)$$

while the other estimates of parameter must satisfy the following equation

$$\sum_{j=1}^{n} \frac{p_q \frac{\partial}{\partial \theta_{qm}} g_q(x_j)}{f(x_j)} = 0 \qquad (2)$$

Equation (1) and (2) will be used to find the estimates of the parameters by expectation maximization (EM) algorithm. The initial estimate of parameters can be found by using respective sample statistics by first setting a suitable threshold point.

3 THE ALGORITHM

Step1:

a) The initial estimates of parameters will be treated as old estimates and substitute into (1) to update the value of $\hat{p}_1$.

b) The updated value of $\hat{p}_1$ together with others initial estimates will be substitute in (2) to revise the value of $\hat{\mu}_1$.

c) The revised $\hat{\mu}_1$ obtained from b) will replace the initial estimates of $\hat{\mu}_1$. This revised $\hat{\mu}_1$ together with others initial estimates will be substituted in (2) again to revise the other values of parameters one at a time until all have been updated.

Step 2:

The estimation process stops if the following are satisfied (otherwise Step 1 will be repeated):

$$\begin{cases} |\hat{p}_1^{new} - \hat{p}_1^{old}| < 0.00001 \\ |\hat{\mu}_1^{new} - \hat{\mu}_1^{old}| < 0.0001 \\ |\hat{\mu}_2^{new} - \hat{\mu}_2^{old}| < 0.0001 \\ |\hat{\sigma}_1^{2(new)} - \hat{\sigma}_1^{2(old)}| < 0.001 \\ |\hat{\sigma}_2^{2(new)} - \hat{\sigma}_2^{2(old)}| < 0.001 \end{cases}$$

The fitted model is checked by performing Komogolov-Smirnov test. The number of pixels in the image is taken to be the number of samples. The large sample size leads to the rejection of the model for all images. The Komogolov-Smirnov statistics lie in the range from 0.0113 to 0.0541 whereas the critical points ranged from 0.00393 to 0.00479.

Graphs of fitted normal mixture model and empirical density function are shown in Fig. 2.

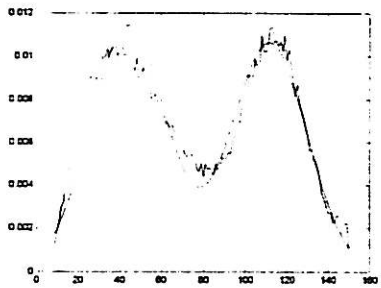


Fig. 2 The line graph of brightness levels(blue) and the fitted model (green) for s101a.jpg.

From the Komogolov-Smirnov statistics we see that the distribution of the model differs from the actual ogive only in small numbers from 0.0113 to 0.05405. This implies that our fitted mixture distribution are good approximation to the data.

4 THE GAMMA-GAMMA MIXTURE MODEL

The probability density function of a gamma distribution $\Gamma(\alpha,\lambda)$ is given by $f(x)=\frac{\lambda^{\alpha}x^{\alpha-1}e^{-\lambda x}}{\Gamma(\alpha)}$, for $x>0$. Hence, the probability density function of the mixture model of two gamma distributions is given by

$$f(x)=\frac{p_1\lambda_1^{\alpha_1}x^{\alpha_1-1}e^{-\lambda_1 x}}{\Gamma(\alpha_1)}+\frac{(1-p_1)\lambda_2^{\alpha_2}x^{\alpha_2-1}e^{-\lambda_2 x}}{\Gamma(\alpha_2)} \qquad -(3)$$

Parameter estimation was performed as before. Fig. 3 shows an example of the fitted model.

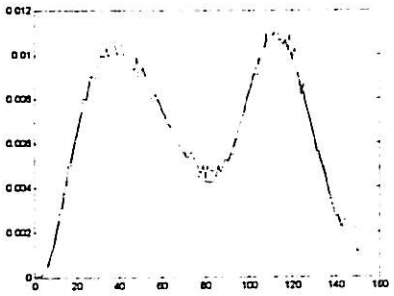


Fig. 3 The line graph of brightness levels (blue) and the fitted gamma mixture model (green) for s101a.jpg.

We observe from the Komogolov-Smirnov statistic that, the gamma models seem to fit certain groups of wire bonding images better than the normal mixture models even though it still suggests rejection of the mixture model at significant level 0.01. For other groups of images, the model shows even bigger deviation from the brightness density compare to the two-component normal mixture models.

5 THE BINARY IMAGE

Having fit a mixture distribution, the threshold point between intermetallics and background is derived [6] using the following rule

$$x \in R_1 \text{ if } \frac{f_1(x)}{f_2(x)} \geq \left(\frac{p_2}{p_1}\right), \text{ otherwise } x \in R_2 \qquad -(4)$$

where $f_j(x)$ is the probability distributions for π_j $(j = 1, 2)$. Equation 4 was applied separately to both cases of normal mixtures and gamma mixtures (i.e. equation 3).

Using a thresholds point, we can set all those pixels with brightness level less than it 0 while all pixels with brightness level higher or equal to it 1. Thus getting a binary image, denoted by BI. The ratio of zeros to the total number of pixels in the binary image BI gives the percentage intermetallic of the wire bond. Fig. 5 shows the original image and its binary counterpart.

The wire bonds that we study are products of different machines. We will label the image according to the machine that produces them, namely, by using alphabet A, B, C, D and E. We find that the gamma mixture model gives larger value of threshold points for most of the images as compare to the normal mixture model. This could be caused by the positively skewed property of the gamma probability density function.

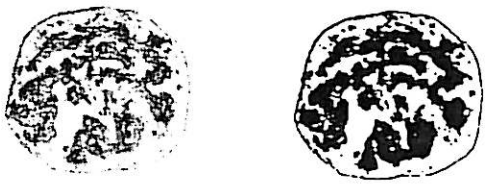


Fig. 5 Original wire bonding image and its binary counterpart.

6 EDGE DETECTION

In Fig. 6, the shaded pixels can be regarded as the neighborhood of the pixel at the center. A pixel is said to be a point on the edge when it has different pixel values with any one of the pixels in its neighborhood. From the binary image BI, we proceed by searching the pixels one by one starting from the upper left corner to the lower right.

Fig 6 Defining the neighborhood of a pixel

Let BI(x, y) represent the brightness level of the (x, y) pixel which is either 1 or 0 in the binary image BI. We will create a new matrix E whose value at the (i, j) location is 0 if and only if a pixel located at (i, j) in BI satisfy

| 2*BI(i, j) - BI(i+1, j) - BI(i, j+1)) | > 0.

The condition above is actually an alternative way to say if any of the values of the pixel BI(i+1, j) or BI(i, j+1) is/are not the same as the value of pixel BI(i, j) then E(i, j) will be marked as boundary. All others pixels will be given value 1.

From the image BI and E that we obtained, we can count the area and perimeter of the intermetallic region. The number of 0 in the BI image will be regarded as the area of the intermetallic region while the number of 0 in the E image will be treated as the length of the perimeter.

7 COMPACTNESS

In pattern recognition, compactness is one of the indicators used to described the feature of an image. Compactness of a region is defined by $\frac{perimeter^2}{4\pi \times area}$. The formula can also be written in the form $\frac{perimeter^2}{\frac{4\pi}{area}}$ where the numerator is the area of circle that can be bounded by a particular perimeter. The compactness is thus represent the ratio of the area of a circle bounded by a particular perimeter and the area of the shape bounded by the same perimeter. It reflects the efficiency in using the perimeter in bounding an object. [7]

A perfect circle has compactness equal to 1, the compactness become bigger and bigger when the image has curly and twisted boundary. Therefore by find the compactness of an image we can measure the degree of 'twist' and uniformity of an intermetallic coverage.

In Figure 7, the compactness measure of two wire bond images is calculated. It is obvious that the image with densely distributed and higher percentage intermetallic produce smaller compactness whereas the one with lower percentage intermetllic and scattered intermetallic coverage gives much higher compactness measure.

(a) (b)

Fig. 7 (a) S201a.jpg, Compactness = 50
(b) S101a.jpg, Compactness = 150.

Figure 8 gives the scatter plot of percentage intermetallic and compactness. The scatter plot above may suggest that when the percentage intermetallic is high, the compactness remains small. However, when percentage intermetallic drops below certain level, the fluctuations can be quite big. It implies that when the percentage intermetallic is low, there can be bigger chance to get a badly bonded wire.

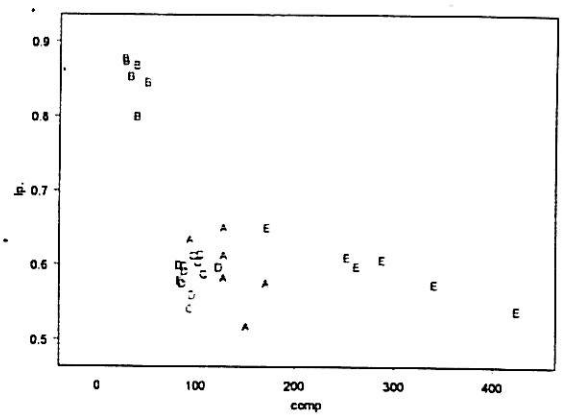


Fig. 8 Scatter plot of percentage intermetallic versus compactness for normal mixture model.

8 RESULTS AND DISCUSSION

It is well established that a high quality wire bond is characterized by high percentage intermetallic (%IM) areas and uniform coverage of intermetallics (low compactness). This study has shown that both %IM and compactness should be simultaneously considered to determine quality of wire-bond. To estimate %IM and compactness, 2-component normal mixtures and 2-component gamma mixtures were fitted. The threshold from gamma mixture tends to be slightly higher than that for the normal mixtures. Nevertheless the %IM from both model were similar in most cases. The threshold derived was used to transform the original image to a binary image from which %IM and compactness were calculated.

9 REFERENCE

[1] Ramsey, T., C. Alfaro and Dowell, H., "Metallurgy's Part in Gold Ball Bonding", *Semiconductor International*, April 1991, pp. 98-102.

[2] Clatterbaugh, G. V., Weiner, J. A. and Charles, H. K., "Gold-Aluminum Intermetallic: Ball Bond Shear Testing and Thin Film Reactions Couples", IEEE Trans. On Components, Hybrids, and Manufacturing Technology, No. 4, 1984, pp. 349-356.

[3] Suresh Kumar, Frank Wulff & Klaus Dittmer, *Degradation of Small Ball Bonds due to Intermetallic Phase(IP) Growth*, Kulicke & Soffa. http://www.kns.com/resources/articles/kumar.pdf

[4] Philofsky, E., Intermetallic Formation in Gold-Aluminum Systems, *Solid State Electronics*, Vol. 13, 1970, 1391-1300.

[5] Everitt, B .S. & H and, D .J. (1981). *Finite Mixture Distributions*. London: Chapman and Hall Ltd.

[6] Richard A. Johnson & Dean W. Wichern (2002), *Applied Multivariate Statistical Analysis*, Fifth Edition, Pearson Education International.

[7] Mark Nixon & Alberto Aguado (2002), *Feature Extraction and Image Processing*, Newnes, United Kingdom.

M. Sc. Statistics
Thesis

A Study of the Au-Al Wire Bonding Image Using Mixture Distribution

by

LIEW KIAN WAH

This project report is submitted in partial satisfaction of the requirements for the degree of

Master of Science in Statistics

UNIVERSITY MALAYA, KUALA LUMPUR

2004

Acknowledgements

I would like to express gratitude to everyone who gave me the possibility to complete this project. Special thanks to my supervisor Associate Professor Dr. Omar Mohd. Rijal for introduce me to this interesting area in studying wire-bonding process. The project is funded by the fundamental research grants, Research Management Centre, UTM. I am indebted to Puan Norliza Mohd. Noor whose experience in image processing for industrial applications has helped me proceeds with the project smoothly. She has also given me a first course in MATLAB programming. My appreciations to Encik Omar Mohd Badar and Texas Instrument (M) Sdn Bhd for help in getting the data sets. I would also like to extend my appreciation to Professor Ong Seng Huat and Professor V. Seshadri for their encouragements and motivations that accelerated the progress. Professor Hsieh Ker-Chang who sent me the article by Philofsky, E. from Taiwan has helped me to clarify doubts over the colours of the intermetallic compounds.

Abstract

The brightness histograms obtained from the transformed Al-Au wire bond images are fitted to two-component normal mixture model and two-component gamma mixture model by finding the maximum likelihood estimates of the parameters. Segmentation of the image is then achieved by determining the threshold points based on the two models by using minimum expected cost method. Minimum error thresholding, which makes use of the Kullback-Leibler divergence, has also been tried on the normal mixture model in finding the threshold points. The binary images defined by these threshold points are then used to estimate the intermetallic percentage of the wire bonds. A measure called 'compactness' is defined to measure the uniformity of the intermetallic coverage of a wire bond due to the importance of uniform coverage in determining the quality of wire bonding.

Abstrak

Histogram kecerahan yang didapatkan daripada gambar cantuman wayar Al-Au akan dimodelkan dengan model campuran dua taburan normal dan model campuran dua taburan gamma. Cara penganggaran kebolehjadian maksimum telah digunakan untuk mencapai tujuan tersebut. Titik-titik ambang yang membezakan intermetalik daripada logam emas terus didapatkan daripada model-model dengan menggunakan cara kos jangkaan minimum dan cara ambang sisihan minimum yang melibatkan pengiraan percapahan Kullback-Leibler. Gambar biner yang ditakrifkan oleh titik-titik ambang tersebut akan digunakan untuk menganggar peratusan intermetalik dalam percantuman wayar. Sukatan 'kepadatan' telah ditakrif dan digunakan sebagai satu ukuran keseragaman liputan intermetalik. Ini adalah kerana keseragaman liputan tersebut adalah penting dalam menentukan kualiti percantuman wayar.

Contents

List of Figures and Graphs

Chapter 1 Introduction

Human eyes are wonderful gifts from the nature. They are extremely strong in visual signal reception. Our brains, on the other hand, are incredibly good in analyzing these signals. It can recognize patterns and differentiate an object from its background easily. The computer, though equipped with high-speed processor, is an infant when visual problems are concerned. The objective of this project is to help engineers in using computer to determine the quality of certain electronic products.

In this chapter, brief introductions to the three main concepts in the project, namely, wire bonding, digital image and finite mixture distributions are presented. We will see how these three objects from different areas be brought together, illustrating a way statistical tools used in tackling engineering problem.

1.1 Wire bonding

Wire bonding is an important process in semiconductor industry. It is an electrical interconnection technique in joining thin wires between electrical device and its conducting track, hence establishing a connection between the inner world of a chip and the world outside. It is used in assembling dynamic random access memory (DRAM) and commodity chips in plastic packages. Two basic wire bonding techniques that are widely used are ball bonding and wedge bonding.

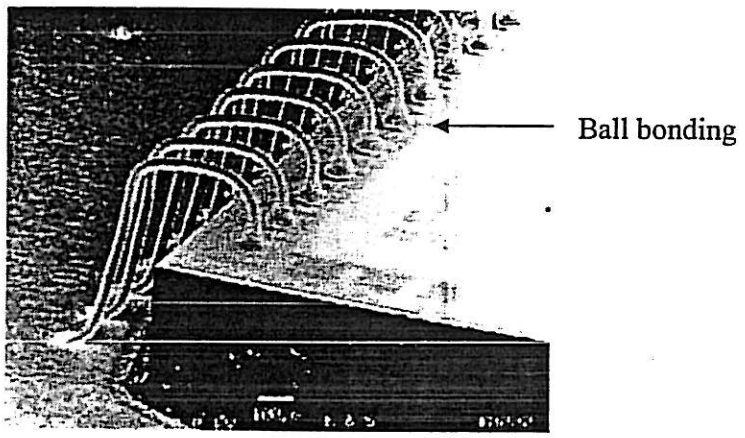


Figure 1.1 25μm gold wire ball/wedge bonds. (Chris Otter, TWI Ltd.)

Information related to wire bonding appears in various resources. Among these, *The Nordic Electronics Packaging Guideline* (2000) gives a comprehensive introduction to the topic.

Gold (Au) and aluminium (Al) interconnections are commonly found in wire bonding process; in which thin gold wires are welded to an aluminium pad under a combination of heat, pressure and/or ultrasonic energy. Wire bond formed as a result of electron sharing and inter-diffusion of atoms when the two metals are brought into intimate contact. The bond strength, which can be measured by performing a shear test, is a benchmark in grading the wire bond.

Although widely used, there are two major drawbacks in the gold-aluminium welding system. One of these is the corrosion initiated by the presence of reactive chemical elements such as halides in the moist environment. The other is the formation of Kirkendall voids as a result of the intermetallic compound growth at high temperature. (Suresh *et al.*)

An intermetallic compound is composed of two or more metallic elements. In the gold-aluminium welding system, five intermetallic compounds: Au_5Al_2, Au_4Al, Au_2Al, AuAl and $AuAl_2$ could be found. However, not all of them can be detected in every wire bond sample. (Ker-Chang Hsieh, 2002) reported that the three major compounds are Au_5Al_2, Au_4Al and Au_2Al.

The intermetallic phase is not static. It may grow thicker and thicker at elevated temperature and the proportions of different intermetallic compounds in the intermetallic layer may also change from time to time. The intermetallic layer is mechanically stronger than the original metals. However, since the rate of Au diffuses into Al is faster than the rate of Al diffuses into Au, voids formed when the bond is exposed to high temperature for long time. These voids are called Kirkendall voids and are always responsible for the peeling of the ball bond.

Due to the delicacy of electronic component, a single broken bond can lead to the failure of the whole part hence jeopardizing the reputation of the manufacturer. In examining the quality of wire bond, Suresh *et al.* found that the bond quality is characterized by the intermetallic coverage in two ways that is, by the percentage of the intermetallic coverage and the uniformity of the coverage. Bonds with uniform intermetallic coverage show stronger bond shear strength than bonds with spotty intermetallic region. Therefore, in order to classify a bond into well-bonded or badly bonded categories, we should examine both characteristics.

In this project, colour images provided by a particular semiconductor production plant will be studied in order to estimate the percentage of intermetallic coverage and to find a way in measuring the uniformity of the coverage. These images were taken by high-powered microscope on the gold balls that were etched from the aluminium pad. Colours of the intermetallic compounds in the images will be used to indicate their existences.

The colours of the intermetallic compounds in the gold-aluminium welding system are shown in Table 1.1. (Philofsky E., 1970)

Intermetallic compound	**Colour**
Au_5Al_2	Tan
Au_4Al	Tan
Au_2Al	Tan
$AuAl$	White
$AuAl_2$	Purple

Table 1.1 Au-Al intermetallic compounds and their corresponding colours

Table 1.1 shows that the colours of the three major phases of the intermetallic coverage, Au_5Al_2, Au_4Al and Au_2Al, are all in tan colour. These colours, which are visually very different from the original gold, also appear in

lower brightness levels in a grey scale image. This property provides us a way to differentiate the gold region from that of the intermetallic region by studying only the grey scale images of the wire bonds.

In Figure 1.2 (a) the original colour image is shown whereas Figure 1.2 (b) shows the grey scale counterpart of the same image transformed by using Matlab. We can see the intermetallic region in the original bond image clearly. In the grey scale image, the intermetallic phase is obviously darker compare to the gold metal.

(a)

(b)

Figure 1.2 (a) Colour image and (b) grey scale image of a wire bond

1.2 Digital image

A digital image is a set of points or picture elements (pixels) with different colours and brightness. When more pixels are used to display an image, the picture is said to have higher resolution and it appears to be smooth and fine. When fewer pixels are used, the resolution is low and the image appears to be blurred.

(a) (b)

Figure1.3 (a) High resolution image and (b) Low resolution image

When a colour photograph is digitized, by using a scanner or other devices, the information is stored in a 3-dimensional array. The number of rows and number of columns in the data array are determined by the resolution of the image. These numbers tell the position of the particular pixel in the image. The information about the colour of a pixel is stored in the three layers corresponding to that position. This is based on the trichromatic theory saying that the visual light spectrum can be decomposed into three primary colours. The three commonly used primary colours are red, green and blue. This concept can be illustrated by Figure 1.4.

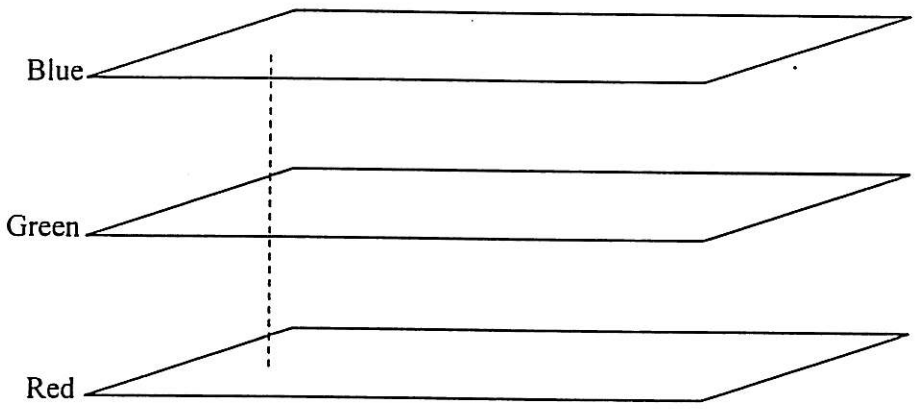


Figure 1.4 Data structure of digital colour image.

In a digital image with RGB encoding scheme, the colour of each pixel is decomposed into right proportion of red (R), green (G) and blue (B) intensities. Each of these intensities is represented by integers from 0 to 255, where 0 represents the weakest intensity and 255 the strongest. For example, the gold colour of the circled region in Figure 1.2 (a) can be decomposed into 254R, 246G and 163B. In the Matlab data structure, the information for red, green and blue intensities are stored in the first, second and third layer respectively.

Another way to study an image is by looking only at he brightness of the pixels. This can be done after the colour image is transformed into a grey scale image. The brightness or luminance of a pixel is characterized by the original colour of that pixel. Analysis of data can be simplified greatly if only the luminance of each pixel is investigated. This is because the luminance level in a grey scale image is stored in a two-dimensional data array rather than three-dimensional data array for a colour image.

The luminance of a pixel, Y can be calculated by using the formula $Y = 0.299R + 0.587G + 0.114B$, where R, G and B are the respective intensities of red, green and blue for that pixel. (John C. Russ, 1994) These coefficients are chosen to match the sensitivity of human's visual perception. By using the formula above, we can calculate the luminance level of the circled region in Figure 1.2 (b), which is 239 when rounded to the nearest integer. The Matlab software can read in a colour image and calculate the luminance of each pixel by using the formula above. When this is done, a grey scale image is obtained and a brightness histogram can be drawn.

In a brightness histogram, the horizontal axis represents the brightness levels while the vertical axis represents the number of pixels in the whole image with corresponding brightness level. We can then calculate the probability of getting certain brightness level by dividing the frequency of that brightness level by the total

number of pixels in that image. Figure 1.5 shows the brightness histogram of the image in Figure 1.2 (b).

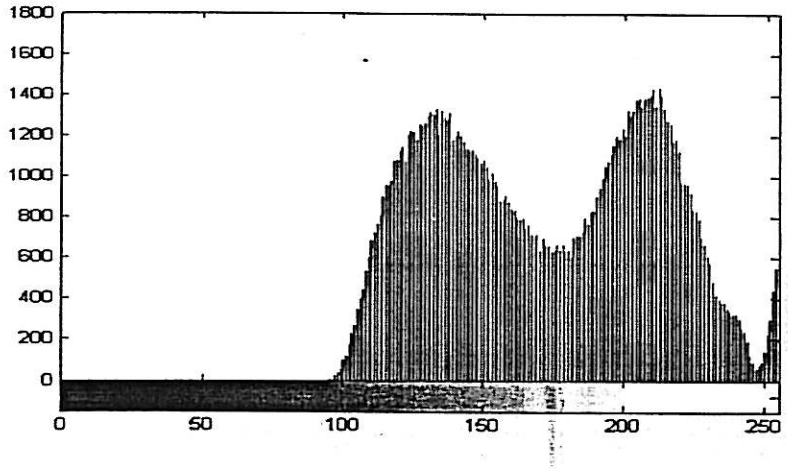


Figure 1.5 Brightness histogram of a wire bond image

Most of the brightness histograms show clearly two humps with one corresponding to the intermetallic region of lower luminance levels and the other corresponding to the gold region of higher luminance levels. The shape of the histogram gives us a hint to model the data by using a mixture of two probability distributions.

1.3 Finite mixture distributions

Many of the commonly used probability density functions are unimodal. It is therefore not suitable to use such models in fitting data that is multimodal or data that is a mixture of two or more distributions; one example is when fitting the brightness histogram. In such cases, mixture distribution model can serve as an alternative choice.

A mixture distribution is a probability distribution with probability density function $f(x)$ of the form $f(x) = \sum_{i=1}^{k} p_i g_i(x)$, where $0 < p_i < 1$ such that $\sum_{i=1}^{k} p_i = 1$

and $g_i(x)$ be probability density function of population π_i. A mixture distribution is said to be finite when k is finite.

Parameter estimation is necessary to specify a mixture distribution. This includes estimation of the proportion parameter p_i and the parameters in each of the component distribution $g_i(x)$. The equations involved in finding the estimator are usually nonlinear; hence numerical method is needed to find the estimates.

(B.S. Everitt & D.J. Hand, 1981) has given a comprehensive description to the method of moments estimation and maximum likelihood estimation that are commonly used in finding the estimates of parameters. The method of moments that involves solving high degree polynomial equations, though can be solved by using a computer, is not adapted in this project due to its sensitivity to the initial values.

The following paragraphs give a description of the maximum likelihood estimation method in its general form. We are going to use these general formulae to derive the maximum likelihood estimators in the following chapters when the component distributions are specified.

Let $x_1, x_2, ..., x_n$ be n independent observations obtained from a population with probability density function $f(x) = \sum_{i=1}^{k} p_i g_i(x)$ and $\tilde{\theta}_i = (\theta_{i1}, \theta_{i2}, ..., \theta_{ih(i)})$ be the parameter vector of $g_i(x)$. In the maximum likelihood estimation, we want to find $p_1, p_2, ..., p_k$ and $\tilde{\theta}_1, \tilde{\theta}_2, ..., \tilde{\theta}_k$ that maximize $\prod_{j=1}^{n} f(x_j)$ or equivalently $\ln\left(\prod_{j=1}^{n} f(x_j)\right)$.

When we want to find the estimates of $p_1, p_2, \ldots, p_k$, the maximization is subject to the constraint $\sum_{i=1}^{k} p_i - 1 = 0$, we introduce the Lagrange multiplier λ and get

$$\nabla \ln\left(\prod_{j=1}^{n} f(x_j)\right) = \lambda \nabla\left(\sum_{i=1}^{k} p_i - 1\right)$$

where the del operator $\nabla = \frac{\partial}{\partial p_1}\underline{v_1} + \frac{\partial}{\partial p_2}\underline{v_2} + \ldots\ldots + \frac{\partial}{\partial p_k}\underline{v_k}$ with $\underline{v_1}, \underline{v_2}, \ldots\ldots, \underline{v_k}$ form a basis of a k-dimensional vector space.

By comparing corresponding component, we get

$$\frac{\partial}{\partial p_i} \ln\left(\prod_{j=1}^{n} f(x_j)\right) = \lambda$$

$$\sum_{j=1}^{n}\left(\frac{\partial}{\partial p_i} \ln f(x_j)\right) = \lambda$$

$$\sum_{j=1}^{n}\left(\frac{1}{f(x_j)} \frac{\partial}{\partial p_i} f(x_j)\right) = \lambda$$

$$\sum_{j=1}^{n}\left(\frac{1}{f(x_j)} \frac{\partial}{\partial p_i} (p_1 g_1(x_j) + p_2 g_2(x_j) + \ldots\ldots + p_k g_k(x_j))\right) = \lambda$$

$$\sum_{j=1}^{n}\left(\frac{g_i(x_j)}{f(x_j)}\right) = \lambda \qquad (1.1)$$

Multiply (1.1) by p_i and sum over i gives

$$\sum_{i=1}^{k}\sum_{j=1}^{n} \frac{p_i g_i(x_j)}{f(x_j)} = \sum_{i=1}^{k} p_i \lambda$$

$$\sum_{j=1}^{n}\sum_{i=1}^{k}\frac{p_i g_i(x_j)}{f(x_j)} = \lambda\sum_{i=1}^{k} p_i$$

$$\lambda = \sum_{j=1}^{n}\sum_{i=1}^{k}\frac{p_i g_i(x_j)}{f(x_j)}$$

$$= \text{n.}$$

Multiplying (1.1) by p_i, we have $\sum_{j=1}^{n}\frac{p_i g_i(x_j)}{f(x_j)} = np_i$. From which we get the maximum likelihood estimator for p_i,

$$\hat{p}_i^{\,new} = \frac{1}{n}\sum_{j=1}^{n}\frac{\hat{p}_i^{\,old} g_i(x_j)}{f(x_j)} \qquad (1.2)$$

Furthermore, by setting $\frac{\partial}{\partial\theta_{qm}}\ln\left(\prod_{j=1}^{n} f(x_j)\right) = 0,$

we have $\frac{\partial}{\partial\theta_{qm}}\left(\sum_{j=1}^{n}\ln f(x_j)\right) = 0$

$$\sum_{j=1}^{n}\left(\frac{\partial}{\partial\theta_{qm}}\ln f(x_j)\right) = 0$$

$$\sum_{j=1}^{n}\left(\frac{1}{f(x_j)}\frac{\partial}{\partial\theta_{qm}}\left(p_1 g_1(x_j) + p_2 g_2(x_j) + \ldots\ldots + p_k g_k(x_j)\right)\right) = 0$$

or $$\sum_{j=1}^{n}\frac{p_q\frac{\partial}{\partial\theta_{qm}} g_q(x_j)}{f(x_j)} = 0 \qquad (1.3)$$

(1.2) and (1.3) will be used to find the estimates of the parameters by a numerical method called expectation maximization (EM) algorithm. Equation (1.3) might seem different from the equations given in (B.S. Everitt & D.J. Hand, 1981) but this general form is applicable to both the mixture model of two normal distributions and the mixture model of two gamma distributions.

The algorithm used in solving (1.2) and (1.3) involves the following steps:

a) **Initial estimate of parameters**: The initial estimate of parameters $p_1, p_2, \ldots, p_k$ and $\tilde{\theta}_1, \tilde{\theta}_2, \ldots, \tilde{\theta}_k$ can be obtained base on the moment estimators or by using Bhattacharya's graphical methods. (B.S. Everitt & D.J. Hand, 1981) Selection of method depends greatly on the mixture components and feasibility in calculations. In this project the moment estimators will be used. However, description and output of the Bhattacharya's method are attached in Appendix A.

b) **Updating new estimates of parameter**: The initial estimates of parameters in step a) are substituted in (1.2) to get updated estimates of the proportion factors. The updated proportion factors are then used in (1.3) to generate new estimates $\tilde{\theta}_1^{(1)}, \tilde{\theta}_2^{(1)}, \ldots, \tilde{\theta}_k^{(1)}$ for $\tilde{\theta}_1, \tilde{\theta}_2, \ldots, \tilde{\theta}_k$. These new estimates are then substituted in (1.2) again to update the estimates of the proportion factors. The updated proportion factors are used in (1.3) again to generate new estimates $\tilde{\theta}_1^{(2)}, \tilde{\theta}_2^{(2)}, \ldots, \tilde{\theta}_k^{(2)}$. These steps are repeated again and again until all corresponding estimates in successive iteration are close enough.

1.4 The project: Past and Present

Early investigation of the wire bonding image by fitting it with mixture distribution can be traced back to Norliza *et al.* (2000), where the writers used a three-component model to fit the brightness histogram but was shown to be a bad fit.

In Norliza *et al.* (2001), the writers improved the estimation by eliminating the background of the image before fitting it to a mixture of two normal distributions.

This project is a continuation of the investigation with slight modifications in both methods and algorithm. The mixture models of two normal distributions and that of two gamma distributions will be studied. Parameters will be estimated using the maximum likelihood estimation and goodness of fit will be checked.

In addition to that I also try to make use of the mixture model in determining the threshold points that differentiates the intermetallic phase from the gold metal. The threshold points obtained can then be used in separating the intermetallic region from the gold region. After the segmentation process, the edge of the intermetallic region can be determined and the perimeter of the intermetallic region can be found. The perimeter will then be used to calculate the 'compactness' of the image that we believe can be used as a measure the uniformity of the intermetallic coverage.

Chapter 2 Pre-processing of the Images

In section 2.1, information about the images of the wire bond will be presented. The semiconductor manufacturing plant from which we obtained the images provides this information. (Omar *et al.*, 1999) Section 2.2 and 2.3 describe a way to clean up the unwanted information in the image before analysis. Pre-processing of image eliminates unwanted noises from the image making it more suitable for analysis.

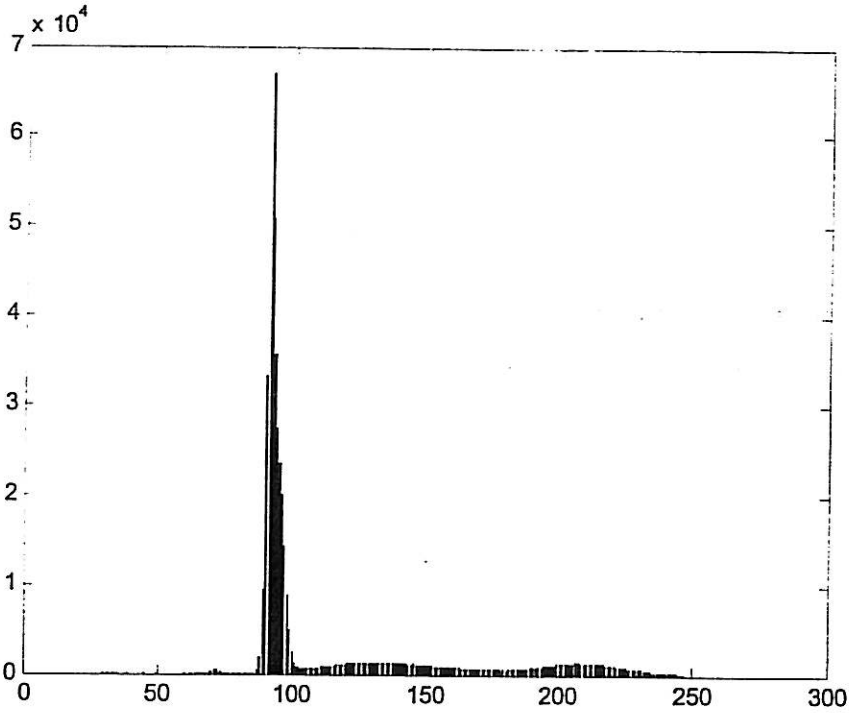
2.1 Obtaining the wire bond images

The data that we are going to analyse are digital images of the wire bonds produced by five different wire bonders. These wire bonds have been etched from the bond pad. A well-bonded sample is put into a 25% Potassium Hydroxide (KOH) solution for 20 to 30 minutes before etching of the bond. This process is to prevent destroy of the intermetallic layer.

Upon removing from the pad, the free ball is then mounted in a tray that allows the welded surface to be orientated perpendicular to the microscope's optical axis. Adjustments are then made to bring the welded surface into focus. A colour video camera was attached to the microscope and a colour video monitor displays the image of the bond at magnification of 500 to 600 times. When the area covered by the intermetallic compound is clearly visible in the screen of the color video monitor, photograph of the bond is taken and saved in Joint Photographic Experts Group (JPEG) format file.

The images obtained from the process above are RGB color images as shown in Figure 2.1 (a). Since our focus will be on the gold ball and the intermetallic region, the dark background of the image must be eliminated before analysis can be carried out. This will eliminate the high peak at the left end of in Figure 2.1 (b).

(a)



(b)

Figure 2.1 (a) The wire bond image with dark background and (b) its brightness histogram.

Norliza *et. al* (2001) reported a way Matlab commands can be used in capturing the region of interest by selecting the points around the gold ball. In this project, we will make use of Adobe Photoshop 6.0.

2.2 Pre-processing of the images by Adobe Photoshop 6.0

Adobe Photoshop is a powerful software in image processing developed by Adobe Systems Incorporated. There is a built in function called magic wand tool, which allow us to select the region of the same or similar colour by clicking the cursor in the region.

We open the image file by using Adobe Photoshop and use the magic wand tool to select dark background region. The selected region will be deleted and replaced by white background when the button 'Delete' is pressed. The white background can be easily eliminated by the Matlab program compare to the dark background that has similar colour to that of the intermetallics.

Usually, the selection takes just a click of the cursor. However, there is time when the intemetallics region is near the circumference of the ball, selection of the background will select the intermetallic region as well. In order to overcome the problem, the pencil tool in Adobe Photoshop is used to draw a short line segment that clear-cut the regions. The effect of this extra line segment to our result is negligible since the short line contains only very few pixels compare to the whole image.

2.3 Pre-processing of the images by Matlab

The processed images obtained from the previous section are now ready for further manipulation. A colour image is stored in Matlab as a three-dimensional array of numbers with the first, second and third layer representing the intensity of red, green and blue respectively. When we convert the image into a grey-scale image,

the luminance level Y of each pixel, which is a number from 0 to 1, is the 'weighted average' given by the equation $Y = \frac{(0.299R + 0.587G + 0.114B)}{255}$.

The grey scale image is stored in a two-dimensional array where each number represents the luminance of the corresponding pixel. The luminance of the image will then grouped into 256 bins according to their brightness level. This can be done by the Matlab command 'imhist'. As a result, we obtain a 256 × 1 matrix with the $(j+1)^{th}$ element representing the number of pixels in the whole image whose luminance level is $\frac{j}{255}$. In order to complete the process, we need to eliminate the white background, which is not a part of the bond, which has luminance level 1.

A technical problem occurs when we try to determine how many luminance levels in the brightness histogram that should be removed. This is because the white background is not represented only by the largest value 1 but appear to be a mixture of brightness level from approximately $\frac{249}{255}$ to 1. This can be verified further by the following experiment.

In the experiment, a picture with pure white background and scribbling of two different colours on it was created. The picture was first saved in Bitmap format, the brightness histogram generated by using Matlab shows exactly three bars with two representing the colour scribble and one representing the white background. When the picture is stored in JPEG format, the brightness histogram generated shows a range of seemingly continuous change of brightness instead of just three bars. The comparison of the artificial pictures and their respective brightness histogram is shown in Figure 2.2.

This phenomenon could be resulting from the compression algorithm used in creating a JPEG file. The compression has caused some blocking effect to the image and mix up the colours.

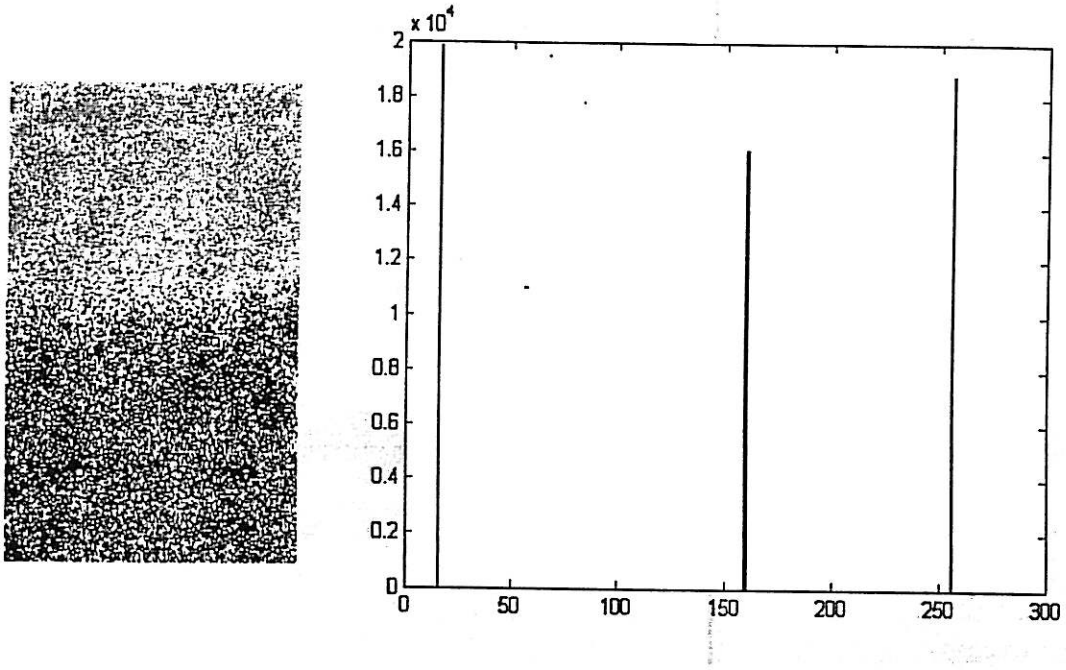


Figure 2.2 a) Artificial image in Bitmap format and its brightness histogram.

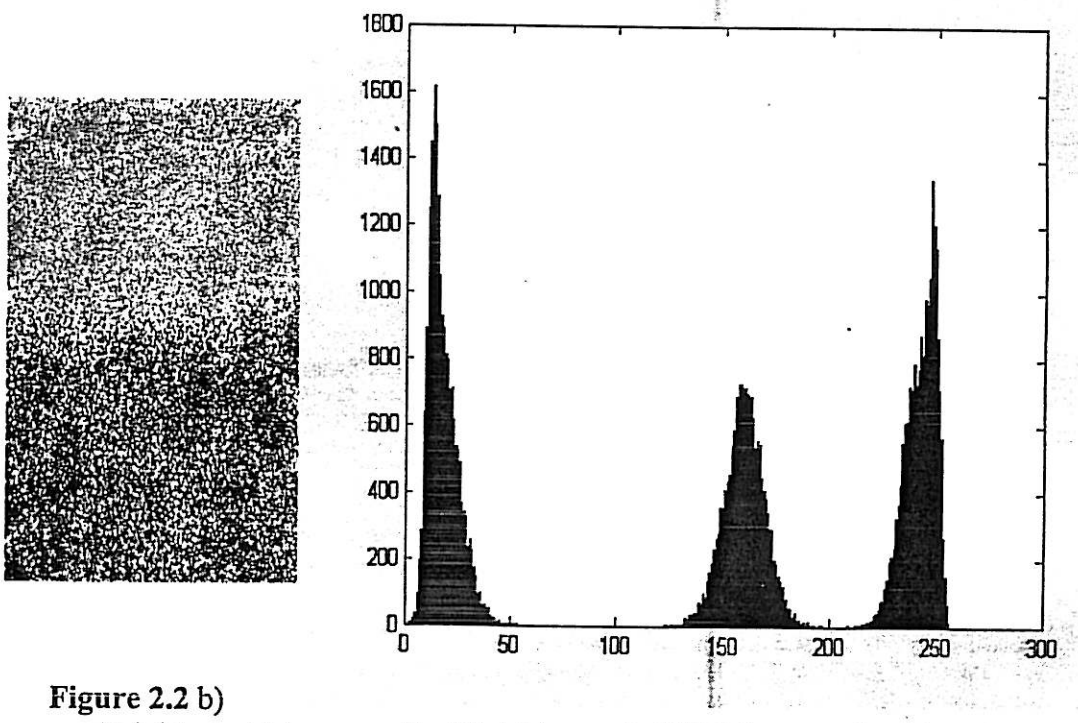


Figure 2.2 b)
Brightness histogram of artificial image in JPEG format when there is no clear cut between two colours of the scribbles.

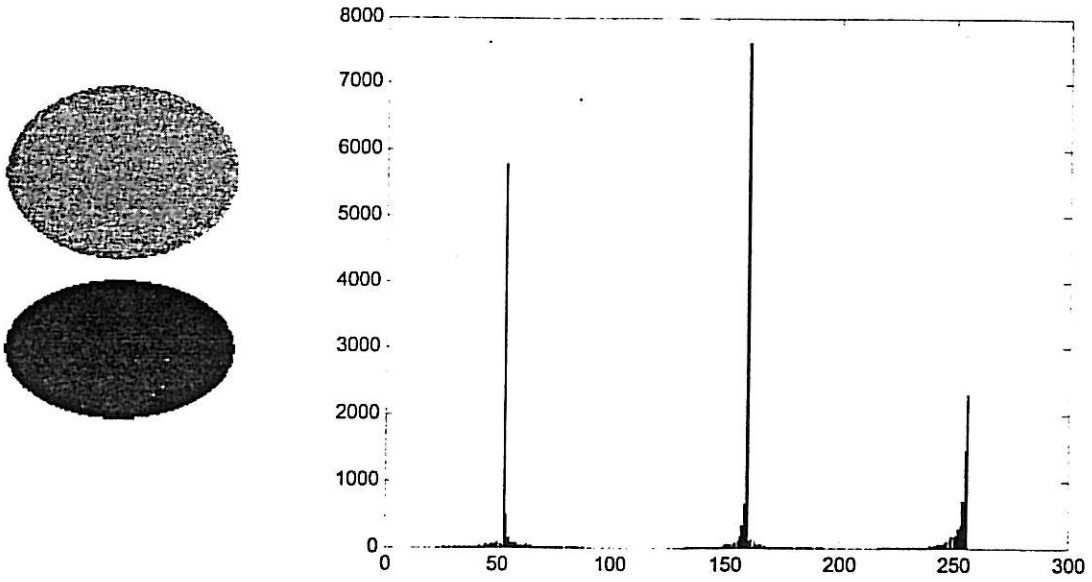


Figure 2.2 c)
Brightness histogram of artificial image with clear-cut edge in JPEG format.

The interference effect is smaller when the boundary between the image and background is clear-cut as shown in Figure 2.2 c).

After studying the brightness histograms of different images, I found that the real image comprises brightness level from about $\frac{96}{255}$ to $\frac{249}{255}$. In order to preserve as much information as possible while carry out the noise elimination process, I will capture the information in the part of the brightness histogram comprising luminance levels $\frac{96}{255}$ to $\frac{249}{255}$ and regarding this as the true image. The range is determined empirically from observations; we believe that this range will not fluctuate too much if the photographs were taken under the same lighting condition.

Using the elimination method described above, we have also translated the brightness histogram to the origin as shown in Figure 2.3 (c). This move has made the fitting of the gamma mixture model easier as the support of the gamma distribution is $x > 0$.

Figure 2.3 (a) The image after the dark background has been deleted

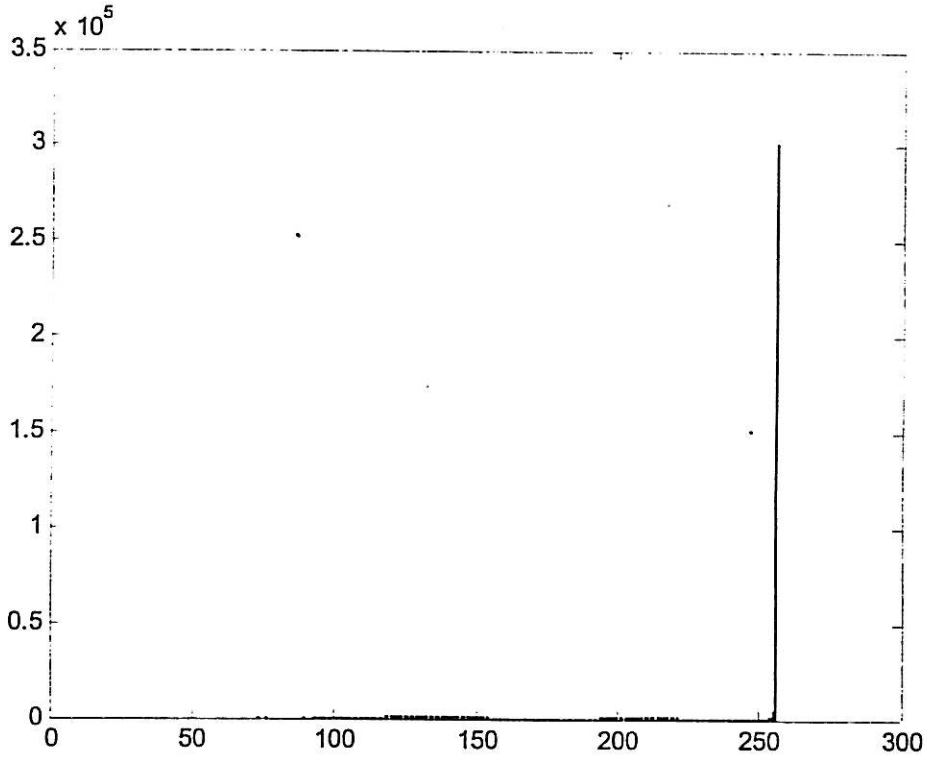


Figure 2.3 (b) The brightness histogram after the dark background has been removed.

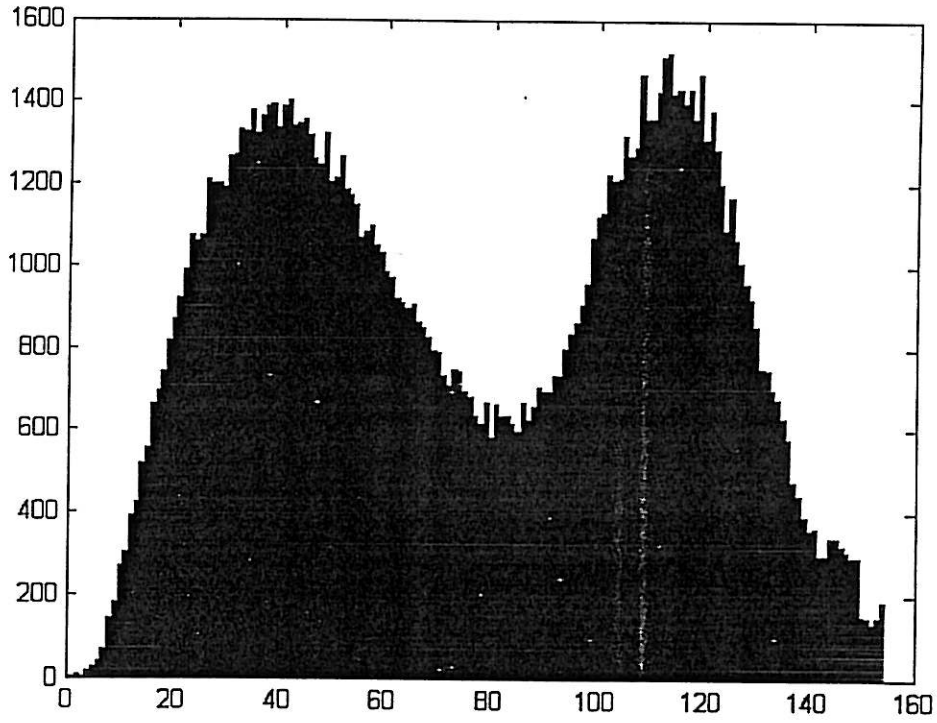


Figure 2.3 (c) Truncated brightness histogram obtained from (b) by capturing only information from luminance levels $\frac{96}{255}$ to $\frac{249}{255}$

Now, we have cleaned up the data and ready for the next step in model building, that is to fit the speculated models to the data.

Chapter 3 Statistical Models for the Brightness Histograms

3.1 The Normal-Normal Mixture Model for the Brightness Histogram

The brightness histograms of the wire bond images usually show clearly two humps that are well separated although sometimes the two humps seems to have merged into one. These brightness histograms have been fitted to a mixture model of two normal distributions in (Norliza *et al*, 2001). The fitted model can then be used to infer the approximate percentage of intermetallic coverage on a specific wire bond.

The probability density function of the mixture of two normal distributions $N(\mu_1,\sigma_1^{\,2})$ and $N(\mu_2,\sigma_2^{\,2})$ in proportion of p_1 and $1-p_1$ respectively can be written explicitly as $f(x)=\frac{p_1}{\sqrt{2\pi\sigma_1^{\,2}}}\exp\left(-\frac{1}{2}\frac{(x-\mu_1)^2}{\sigma_1^{\,2}}\right)+\frac{1-p_1}{\sqrt{2\pi\sigma_2^{\,2}}}\exp\left(-\frac{1}{2}\frac{(x-\mu_2)^2}{\sigma_2^{\,2}}\right)$.

In order to estimate the parameters in the mixture model, initial estimates of the parameters must be made. These initial estimates of parameters will then be used in the EM algorithm, which repeatedly updates the values of the estimates until convergence attained.

3.1.1 Initial Estimates of Parameters

As described by (B.S.Everitt & D.J.Hand, 1981), the maximum likelihood estimates of the parameters are less sensitive to the initial estimates compare to the method of moment. This property gives us bigger freedom in finding the initial estimates.

Initial estimates of the parameter can be done in various ways. In a two-component univariate normal mixture model, we can use the moment estimators or graphical technique suggested by Bhattacharya for this purpose. The moment estimators will be used in this project due to its simplicity in addition to its easily

programmable property. However, the Bhattacharya graphical method has also been tried and the results are shown in Appendix A.

3.1.2 Selecting the initial threshold point

First of all we need to transform the brightness histogram such that the vertical axis representing the probability of getting a particular brightness level. This is done by dividing the number of pixels in each brightness level by the total number of pixels in the wire bond image.

Secondly, we set a threshold point such that all the luminance levels below it represent the intermetallic phase while pixels with larger luminance represents the gold metal. A straightforward selection of the threshold point is the minimum point between the two humps of the brightness histogram. We can input the initial threshold point by inspecting the brightness histogram.

After trying different input threshold values on the same image, I found that the final parameters estimated do not seem to be influenced much by the value of the threshold point within certain range when the number of iterations is big enough. The results are tabulated in Table 3.1, where three images with different features are tested by input different cutting points. We can see that, unless the input threshold is at the far left or right end, there is little change in the final results other than the number of iterations.

Therefore, for simplicity, we can use a fixed initial estimated threshold in the programme and apply it to all images. From Table 3.1 we also observe that if the initial threshold is either too big or too small, the number of iterations tends to be big or the estimation will become inaccurate. Therefore, we will use $\frac{180}{255}$ as our default initial threshold in the programme. Another reason this can be done is because the brightness level that differentiates the gold metal from the intermetallic compounds should always lie within certain luminance range if the photographs are taken under the same lighting condition.

Input threshold	No of Iterations	p	mu1	sigma1	mu2	sigma2	KS-statistic
15	93	0.5185	44.02	328.58	115.48	328.26	0.0233
40	71	0.5185	44.02 .	328.58	115.48	328.26	0.0233
100	58	0.5186	44.02	328.63	115.48	328.20	0.0233
135	80	0.5186	44.02	328.63	115.48	328.20	0.0233
146	99	0.9977	76.34	1456.37	145.24	2.76	0.0897

(a) s101a.jpg (Well separated humps and about moderate percentage intermetallic)

Input threshold	No of Iterations	p	mu1	sigma1	mu2	sigma2	KS-statistic
15	317	0.8838	62.38	646.09	130.70	100.29	0.0272
40	228	0.8838	62.38	646.09	130.70	100.29	0.0272
100	210	0.8838	62.38	646.12	130.70	100.27	0.0272
135	39	0.8838	62.38	646.12	130.70	100.27	0.0272
146	46	0.8838	62.38	646.12	130.70	100.27	0.0272

(b) s201a.jpg (Well separated humps with high percentage intermetallic)

Input threshold	No of Iterations	p	mu1	sigma1	mu2	sigma2	KS-statistic
15	1080	0.7026	72.35	682.51	112.02	235.64	0.0126
40	1031	0.7026	72.35	682.50	112.02	235.64	0.0126
100	296	0.7026	72.35	682.50	112.02	235.64	0.0126
135	695	0.7041	72.41	683.71	112.06	234.38	0.0125
146	98	1	84.15	878.42	148	0	0.0276

(c) s503a.jpg (Not well separated humps with high percentage intermetallic)

Table 3.1 The effect of different input threshold values to the estimation of parameters

3.1.3 Initial estimates of proportion factors, means and variances

After selecting the threshold point, a brightness histogram is divided into two regions, those with lower luminance will be treated as the intermetallic region while those with higher luminance will be treated as gold metal. The initial estimate of others parameters can be done in the following steps:

a) **Estimating p_1:** The proportion of intermetallic will be estimated by: $\hat{P}_1 = \sum_{x \le \text{threshold}} p_x$, that is the sum of all probabilities of luminance levels less than or equal to the threshold point.

b) **Estimating μ_1 and μ_2:** The brightness levels with the highest probability in intermetallic region and gold region will be used as respective initial estimate of μ_1 and μ_2.

c) **Estimating σ_1^2 and σ_2^2:** The average of square of distance from the mean will be used in finding the initial estimates of σ_1^2 and σ_2^2, writing it explicitly we have

$$\hat{\sigma}_1^{\,2} = \frac{\sum_{x \le \text{threshold}} [(x - \hat{\mu}_1)^2 \times n_x]}{\text{Total number of pixels in the intermetallic region}}$$

and $$\hat{\sigma}_2^{\,2} = \frac{\sum_{x > \text{threshold}} [(x - \hat{\mu}_2)^2 \times n_x]}{\text{Total number of pixels in the gold region}},$$

where n_x is the number of pixels with luminance level x.

3.1.4 The EM Algorithm

In the model of the mixture of two Gaussian distributions, where

$$f(x) = \frac{p_1}{\sqrt{2\pi\sigma_1^{\,2}}} \exp\left(-\frac{1}{2}\frac{(x-\mu_1)^2}{\sigma_1^{\,2}}\right) + \frac{1-p_1}{\sqrt{2\pi\sigma_2^{\,2}}} \exp\left(-\frac{1}{2}\frac{(x-\mu_2)^2}{\sigma_2^{\,2}}\right),$$

not all of the maximum likelihood estimators of the parameters are explicitly solvable. However, we can use the iterative Expectation-Maximization (EM) Algorithm to solve for the estimates numerically.

The idea behind the EM algorithm is simple. Initial estimates of parameters should first be obtained. In (1.2), we have seen that

$$\hat{p}_i^{\,new} = \frac{1}{n}\sum_{j=1}^{n}\frac{\hat{p}_i^{\,old}\, g_i(x_j)}{f(x_j)}$$

is a maximum likelihood estimator for the proportion parameter p_1. By taking

$$g_1(x) = \frac{1}{\sqrt{2\pi\sigma_1^{\,2}}}\exp\left(-\frac{1}{2}\frac{(x-\mu_1)^2}{\sigma_1^{\,2}}\right)$$

we get

$$\hat{p}_1^{\,new} = \frac{1}{n}\sum_{j=1}^{n}\frac{\frac{\hat{p}_1^{\,old}}{\sqrt{2\pi\sigma_1^{\,2}}}\exp\left(-\frac{1}{2}\frac{(x_j-\mu_1)^2}{\sigma_1^{\,2}}\right)}{f(x_j)} \qquad (3.1)$$

as the estimator for the first proportion parameter.

Similarly by putting $\theta_{qm} = \mu_1, \mu_2, \sigma_1^{\,2}$ and $\sigma_2^{\,2}$ respectively in (1.3), that is

$$\sum_{j=1}^{n}\frac{p_q\frac{\partial}{\partial\theta_{qm}}g_q(x_j)}{f(x_j)} = 0,$$

we can get the conditions (3.2) to (3.5) that the MLE must satisfy. Details of the derivation of equations (3.2) to (3.5) are given in Appendix B.

$$\hat{\mu}_1^{new} = \frac{\sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j - \hat{\mu}_1^{old})^2}{\sigma_1^2}\right) x_j}{f(x_j)}}{\sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j - \hat{\mu}_1^{old})^2}{\sigma_1^2}\right)}{f(x_j)}} \tag{3.2}$$

$$\hat{\mu}_2^{new} = \frac{\sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j - \hat{\mu}_2^{old})^2}{\sigma_1^2}\right) x_j}{f(x_j)}}{\sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j - \hat{\mu}_2^{old})^2}{\sigma_1^2}\right)}{f(x_j)}} \tag{3.3}$$

$$\hat{\sigma}_1^{2(new)} = \frac{\sum_{j=1}^{n} \frac{p_1 \exp\left(-\frac{1}{2}\frac{(x_j - \mu_1)^2}{\hat{\sigma}_1^{2(old)}}\right)(x_j - \mu_1)^2}{f(x_j)}}{\sum_{j=1}^{n} \frac{p_1 \exp\left(-\frac{1}{2}\frac{(x_j - \mu_1)^2}{\hat{\sigma}_1^{2(old)}}\right)}{f(x_j)}} \tag{3.4}$$

and

$$\hat{\sigma}_2^{2(new)} = \frac{\sum_{j=1}^{n} \frac{(1-p_1)\exp\left(-\frac{1}{2}\frac{(x_j - \mu_2)^2}{\hat{\sigma}_2^{2(old)}}\right)(x_j - \mu_2)^2}{f(x_j)}}{\sum_{j=1}^{n} \frac{(1-p_1)\exp\left(-\frac{1}{2}\frac{(x_j - \mu_2)^2}{\hat{\sigma}_2^{2(old)}}\right)}{f(x_j)}} \tag{3.5}$$

The EM algorithm flows as follow:

Step1:

a) The initial estimates of parameters will be treated as old estimates and substitute into (3.1) to update the value of $\hat{p}_1$.

b) The updated value of $\hat{p}_1$ will replace the initial estimates of $\hat{p}_1$. This together with others initial estimates will be substitute in (3.2) to revise the value of $\hat{\mu}_1$.

c) The revised $\hat{\mu}_1$ obtained from b) will replace the initial estimates of $\hat{\mu}_1$. This revised $\hat{\mu}_1$ together with others initial estimates will be substituted in (3.3) to revise the value of $\hat{\mu}_2$.

d) The steps continue by using (3.4) and (3.5) to find the updated values for $\hat{\sigma}_1^2$ and $\hat{\sigma}_2^2$.

Step 2:

The revised estimates obtained in Step 1 will be checked to find out if they are close to the previous estimation. If all successive estimates are close enough, the process stops or else Step 1 will be repeated again but using the set of revised values as old estimates. In this project, the conditions to terminate the algorithms are:

$$\begin{cases} |\hat{p}_1^{new} - \hat{p}_1^{old}| < 0.00001 \\ |\hat{\mu}_1^{new} - \hat{\mu}_1^{old}| < 0.0001 \\ |\hat{\mu}_2^{new} - \hat{\mu}_2^{old}| < 0.0001 \\ |\hat{\sigma}_1^{2(new)} - \hat{\sigma}_1^{2(old)}| < 0.001 \\ |\hat{\sigma}_2^{2(new)} - \hat{\sigma}_2^{2(old)}| < 0.001 \end{cases}$$

Table 3.2 shows the tabulated results of the estimated parameters. Figure 3.1 (a) to Figure 3.5 (a) show the graphs of the brightness levels together with the fitted models for images from different wire bonders. Figure 3.1 (b) to Figure 3.5 (b) show the graphs of the empirical frequency distribution (EFD) drawn from the respective brightness histogram and the graph of the distribution function of the fitted mixture model.

Image	Initial estimates of p	mu1	sigma1	mu2	sigma2	Final estimates of p	mu1	sigma1	mu2	sigma2
S101a	0.5637	135	423	207	237	0.5146	137.85	324.3	205.45	341.4
S102a	0.6786	123	512	201	197	0.5758	128.13	197.2	195.21	401.8
S103a	0.6463	136	374	213	244	0.6060	138.06	308.8	204.99	334.8
S104a	0.5856	134	496	219	267	0.6397	149.47	496.6	219.77	179.3
S105a	0.6533	132	475	202	225	0.5683	136.62	301.5	198.64	388.2
S106a	0.5992	144	367	217	296	0.6501	149.00	493.9	218.21	233.2
S201a	0.7366	168	772	182	1116	0.8748	156.29	644.1	225.58	129.3
S202a	0.7928	144	239	236	663	0.8000	144.12	257.8	219.15	341.2
S203a	0.8102	158	368	244	1130	0.8533	151.26	360.7	224.32	226.3
S204a	0.8242	162	383	182	1540	0.8798	154.77	366.8	226.64	184.5
S205a	0.8470	154	402	221	386	0.8730	145.84	362.8	219.14	257.1
S206a	0.8622	142	269	211	340	0.8378	138.18	238.1	205.77	537.0
S301a	0.6658	118	620	223	315	0.6099	129.07	291.3	208.69	459.8
S302a	0.6900	115	558	212	192	0.6003	122.81	174.1	198.55	460.8
S303a	0.6555	121	473	212	192	0.5966	126.34	242.7	203.30	346.7
S304a	0.6460	108	730	205	184	0.5353	116.48	144.0	196.07	504.8
S305a	0.6898	111	569	208	177	0.5808	116.45	133.2	194.37	541.6
S306a	0.6695	120	544	219	261	0.6094	128.54	266.7	207.08	449.7
S401a	0.6935	111	542	211	238	0.5679	116.47	95.0	196.49	840.1
S402a	0.7111	111	581	213	208	0.5861	118.39	108.9	194.71	737.2
S403a	0.6692	113	560	213	224	0.5515	117.84	120.2	198.13	691.2
S404a	0.6916	109	629	213	216	0.5950	118.21	146.2	199.05	595.2
S405a	0.6739	107	665	223	306	0.5851	117.30	154.0	204.52	687.3
S406a	0.6981	111	526	215	230	0.5661	114.64	85.6	194.72	843.9
S501a	0.4937	177	918	213	310	0.5784	159.92	565.8	210.88	330.0
S502a	0.4769	178	908	186	604	0.5680	162.91	640.7	205.21	303.2
S503a	0.5190	181	1042	185	589	0.6257	163.49	643.0	203.72	314.8
S504a	0.5096	167	590	210	263	0.6141	160.27	629.0	209.95	281.6
S505a	0.6266	160	435	183	741	0.5943	152.14	457.0	199.69	436.0
S506a	0.5883	161	551	209	269	0.6444	152.78	553.4	210.26	293.2

Table 3.2 Initial and final estimates of parameters in the normal mixture model.

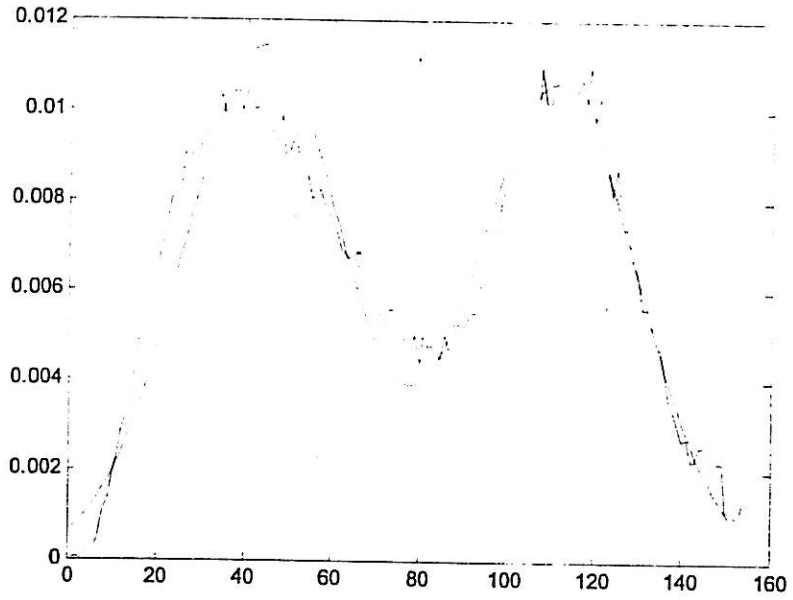


Figure 3.1 (a) The line graph of brightness levels (blue) and the fitted model (green) for s101a.jpg

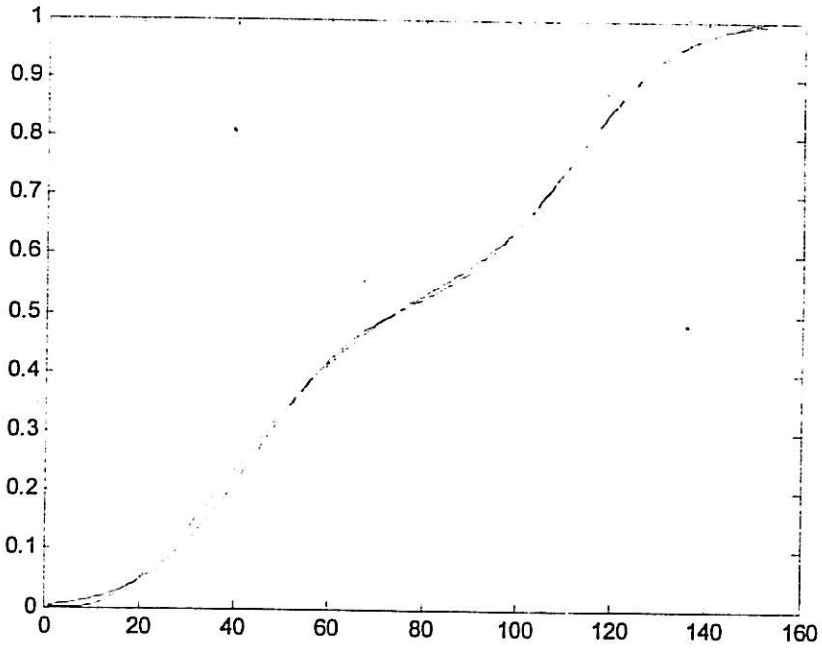


Figure 3.1 (b) The EFD of brightness levels (blue) and the fitted distribution function (green) for s101a.jpg

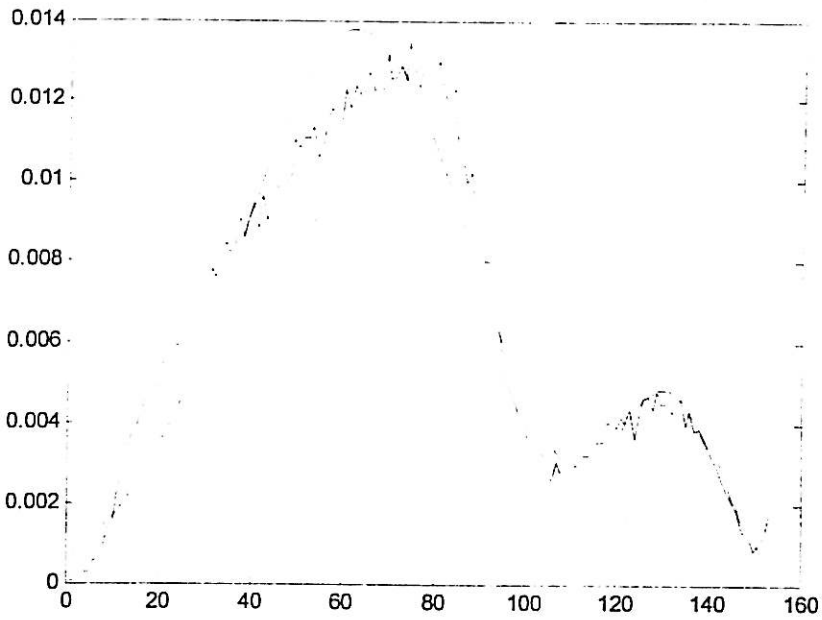


Figure 3.2 (a) The line graph of brightness levels (blue) and the fitted model (green) for s201a.jpg

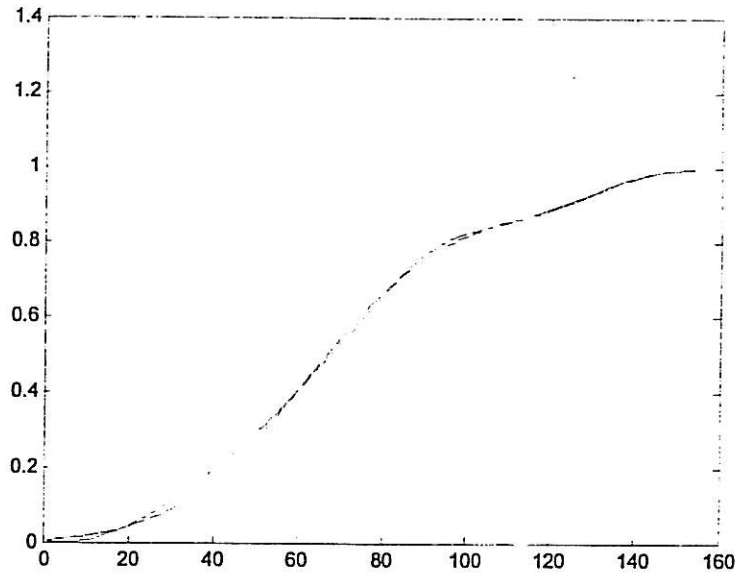


Figure 3.2 (b) The EFD of brightness levels (blue) and the fitted distribution function (green) for s201a.jpg

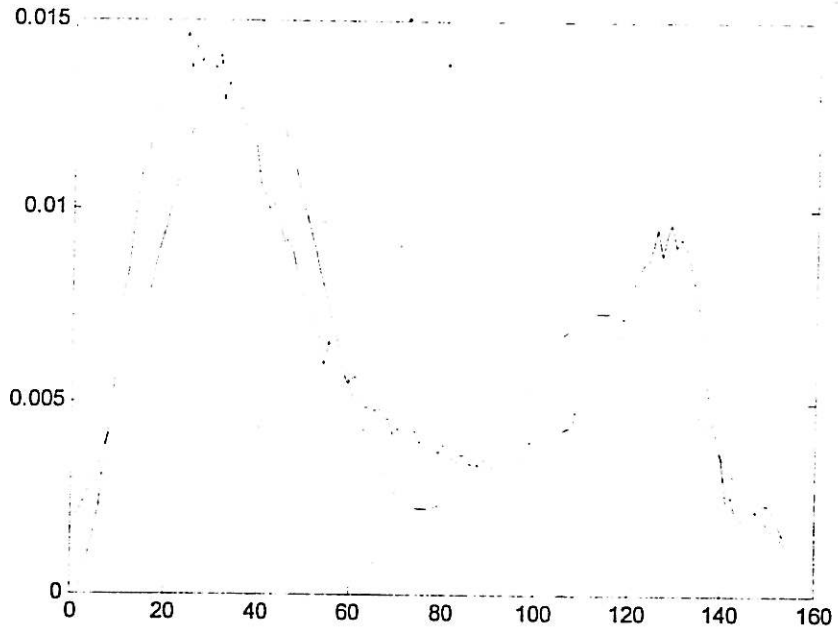


Figure 3.3 (a) The line graph of brightness levels (blue) and the fitted model (green) for s301a.jpg

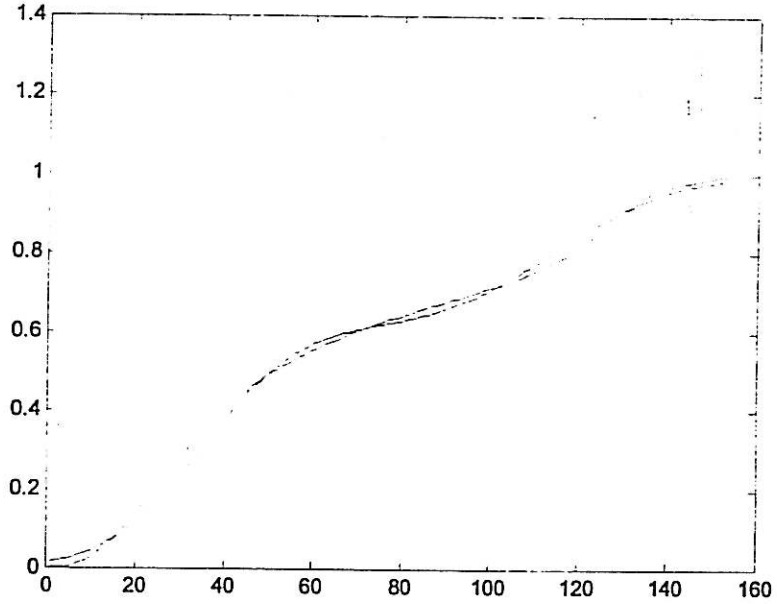


Figure 3.3 (b) The EFD of brightness levels (blue) and the fitted distribution function (green) for s301a.jpg

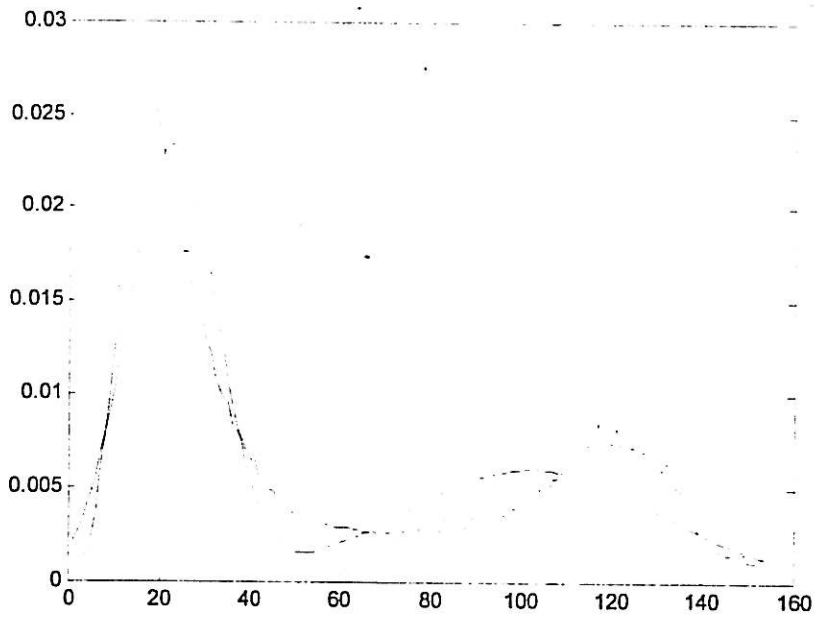


Figure 3.4 (a) The line graph of brightness levels (blue) and the fitted model (green) for s401a.jpg

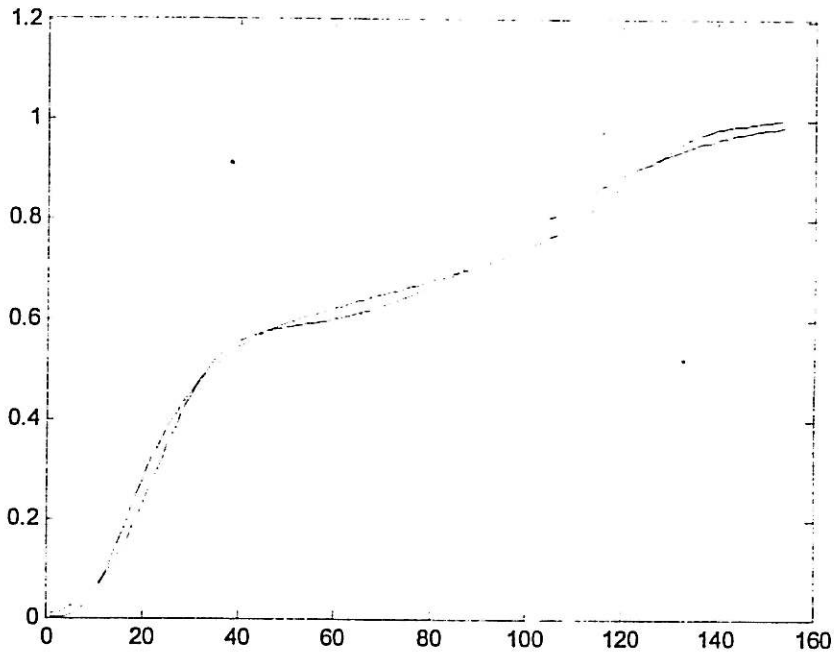


Figure 3.4 (b) The EFD of brightness levels (blue) and the fitted distribution function (green) for s401a.jpg

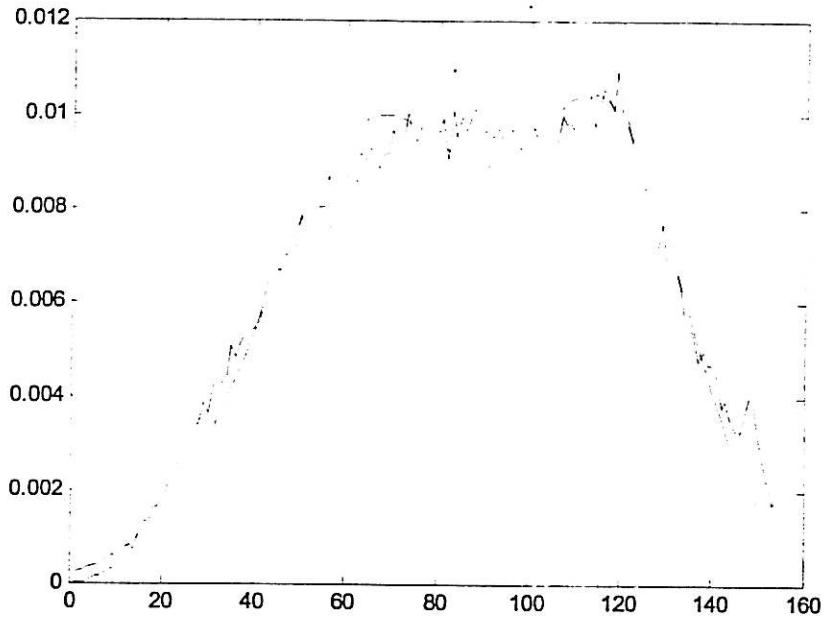


Figure 3.5 (a) The line graph of brightness levels (blue) and the fitted model (green) for s501a.jpg

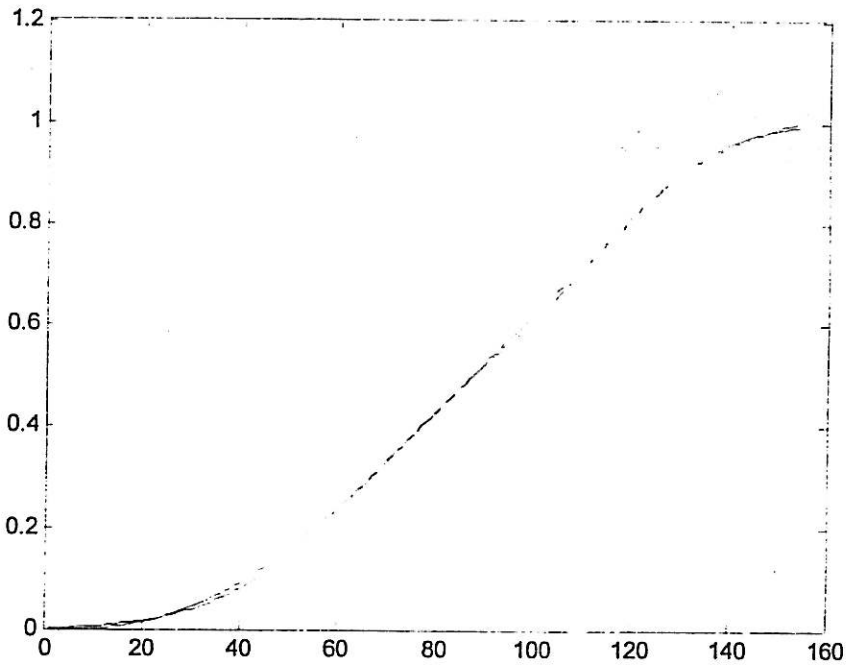


Figure 3.5 (b) The EFD of brightness levels (blue) and the fitted distribution function (green) for s501a.jpg

3.1.5 Model Checking

After getting the estimates of the parameter for the mixture model, we can now specify the model. A question remains is whether the model really describes our data well. One way to answer the question is by performing the goodness of fit test. The Komogolov-Smirnov test will be used in the project.

The Komogolov-Smirnov test is based on the difference between a predetermined distribution function specified in the null hypothesis and the empirical distribution function.

Let $x_1, x_2, ..., x_n$ be observed samples used to construct the empirical distribution function F_n(x), which is defined by $F_n(x) = \frac{i}{n}$ such that exactly i of the observed samples are less than or equal to x. F_n(x) is calculated by finding the cumulative sum of probability of getting a pixels with brightness level less than or equal to x.

Suppose F_0(x) is the hypothesized distribution function. The Kolmogolov-Smirnov statistic D_n, which is defined by $D_n = \sup_{-\infty<x<\infty} |F_n(x) - F_0(x)|$ is a measurement of the maximum deviation of the empirical distribution function from the hypothesized distribution function. Therefore, we should reject H_0 if D_n appears to be large.

In our case, we will test whether the brightness of the pixels in the wire bond image can be regarded as samples from the estimated mixture model.

Therefore, we have

$H_0 : F = \hat{F}_0(x)$, where

$$\hat{F}_0(x) = \int_{-\infty}^{x} \left(\frac{\hat{p}_1}{\sqrt{2\pi\hat{\sigma}_1^{\,2}}} \exp\left(-\frac{1}{2} \frac{(x-\hat{\mu}_1)^2}{\hat{\sigma}_1^{\,2}} \right) + \frac{1-\hat{p}_1}{\sqrt{2\pi\hat{\sigma}_2^{\,2}}} \exp\left(-\frac{1}{2} \frac{(x-\hat{\mu}_2)^2}{\hat{\sigma}_2^{\,2}} \right) \right) dx$$

and $H_1 : F \neq \hat{F}_0(x)$.

$D_n = \sup_{-\infty<x<\infty} |F_n(x) - \hat{F}_0(x)|$ will be used as the test statistic. In the wire bond image, the number of samples is the number of pixels in the image. This is usually a large number and we cannot find the exact critical point from the statistical table. Therefore, D_n will be compared to $\frac{1.63}{\sqrt{n}}$ which gives us a test at significant level 0.01 (J. Murdoch, 1998).

Table 3.3 shows the results of the calculated Komogolov-Smirnov statistics and the corresponding critical points. The critical points are changing because of the different sizes of the actual wire bond image after deleting the background. We can also see that, due to the large sample size in the study, the critical points of the test are relatively small compare to the Komogolov-Smirnov statistics calculated. The Komogolov-Smirnov statistics lie in the range from 0.0113 to 0.0541 whereas the critical points ranged from 0.00393 to 0.00479.

From the discussion above, we are forced to reject the hypothesis, which means that the brightness of the pixels in the wire bond images should not be regarded as samples from a distribution of the mixture of two normal distributions with the estimated parameters.

A weakness of this test is that, since we cannot specify the mixture model precisely as we do not know any of the parameters, the distribution function in the null hypothesis contains estimates of parameters from the sample itself. The stochastic behavior of the estimators gives extra uncertainties in the model. Therefore, we have a conservative test. A conservative test is a test that has smaller type I error but bigger type II error. In other words, since we get the hypothesized model from the samples, we have a greater chance of failing to reject H_0 when in fact the alternative hypothesis is true.

No	Image	KS-stat	Critical Pt.
1	s101a	0. 0231	0. 00445
2	s102a	0. 0303	0. 00448
3	s103a	0. 0262	0. 00446
4	s104a	0. 0268	0. 00444
5	s105a	0. 0241	0. 00422
6	s106a	0. 0262	0. 00411
7	s201a	0. 0270	0. 00478
8	s202a	0. 0166	0. 00479
9	s203a	0. 0177	0. 00453
10	s204a	0. 0229	0. 00460
11	s205a	0. 0214	0. 00470
12	s206a	0. 0168	0. 00440
13	s301a	0. 0424	0. 00404
14	s302a	0. 0457	0. 00404
15	s303a	0. 0383	0. 00401

No	Image	KS-stat	Critical Pt.
16	s304a	0. 0466	0. 00403
17	s305a	0. 0473	0. 00407
18	s306a	0. 0395	0. 00405
19	s401a	0. 0494	0. 00409
20	s402a	0. 0478	0. 00395
21	s403a	0. 0438	0. 00400
22	s404a	0. 0541	0. 00408
23	s405a	0. 0520	0. 00415
24	s406a	0. 0451	0. 00393
25	s501a	0. 0126	0. 00417
26	s502a	0. 0131	0. 00427
27	s503a	0. 0113	0. 00417
28	s504a	0. 0127	0. 00423
29	s505a	0. 0151	0. 00433
30	s506a	0. 0213	0. 00406

Table 3.3 Comparison of Komogolov-Smirnov statistics and corresponding critical points for the two-component normal mixture model

Although the hypothesized model did not fit our data well enough from the statistical point of view, we can still make use of the model in some ways. From the Komogolov-Smirnov statistics we see that the distribution of the model differs from the actual ogive only in small numbers from 0.0113 to 0.05405. This implies that if we use the model to estimate the intermetallic percentage at a certain threshold point, our maximum percentage error in the estimation will be around 1% to 5.4%, which is quite satisfactory compare to the traditional way of estimation by using human eyes.

3.2 The Gamma-gamma Mixture Model for the Brightness Histogram

As we have seen in the previous section, the normal-normal mixture model did not fit the data well. If we look at some of the brightness histogram carefully, we might suspect that the component distribution is not symmetrical. Therefore, in this section, a mixture model of two gamma components will be tried to fit the brightness histogram.

The gamma mixture model has been used widely in fitting the model of grey level image obtained from various sources, for example: in the analysis of Synthetic Aperture Radar (SAR) image. (Ali El Zaart *et al*, 1999).

The probability density function of a gamma distribution $\Gamma(\alpha,\lambda)$ is given by $f(x)=\frac{\lambda^{\alpha}x^{\alpha-1}e^{-\lambda x}}{\Gamma(\alpha)}$, for $x>0$. Hence, the probability density function of the mixture model of two gamma distributions is given by

$$f(x)=\frac{p_1\lambda_1{}^{\alpha_1}x^{\alpha_1-1}e^{-\lambda_1 x}}{\Gamma(\alpha_1)}+\frac{(1-p_1)\lambda_2{}^{\alpha_2}x^{\alpha_2-1}e^{-\lambda_2 x}}{\Gamma(\alpha_2)}.$$

Similar to what we did in fitting the normal mixture model, we will use the truncated histogram as the 'real image' for modeling. However, care must also be taken to avoid the inclusive of zero frequency in the histogram, which will generate 'log of zeros' error in the estimation. If the histogram in the brightness levels $\frac{96}{255}$ to $\frac{249}{255}$ contain level with frequency zero, which might happen at the lower end, then we have to modify the captured range a bit by shifting it one or two levels higher that is from $\frac{97}{255}$ to $\frac{250}{255}$ or from $\frac{98}{255}$ to $\frac{251}{255}$.

The execution of the gamma mixture algorithm is far more complicated than that of the normal mixture model. This is partly due to the presence of logarithm, gamma and digamma function in the estimators.

Applying $\hat{p}_i^{new} = \frac{1}{n}\sum_{j=1}^{n} \frac{\hat{p}_i^{old} g_i(x_j)}{f(x_j)}$ to update the estimate of $\hat{p}_1$ is straightforward, that is by using

$$\hat{p}_1^{new} = \frac{1}{n}\sum_{j=1}^{n} \frac{\hat{p}_1^{old} \lambda_1^{\alpha_1} x_j^{\alpha_1 - 1} e^{-\lambda_1 x_j}}{\frac{\Gamma(\alpha_1)}{f(x_j)}} \tag{3.6}$$

as the estimator for the first proportion parameter.

Finding the estimates of $\alpha_1, \lambda_1, \alpha_2$ and λ_2, however, takes some effort by using (1.3) again. Details of the derivation of the following equations are given in Appendix C. From Appendix C we can see that the maximum likelihood estimators of $\alpha_1, \lambda_1, \alpha_2$ and λ_2 must satisfy the following conditions.

$$\hat{\lambda}_1^{new} = \frac{\sum_{j=1}^{n} \frac{x_j^{\alpha_1 - 1}\left(\alpha_1 e^{-\hat{\lambda}_1^{old} x_j}\right)}{f(x_j)}}{\sum_{j=1}^{n} \frac{x_j^{\alpha_1} e^{-\hat{\lambda}_1^{old} x_j}}{f(x_j)}} \tag{3.7}$$

$$\hat{\lambda}_2^{new} = \frac{\sum_{j=1}^{n} \frac{x_j^{\alpha_2 - 1}\left(\alpha_2 e^{-\hat{\lambda}_2^{old} x_j}\right)}{f(x_j)}}{\sum_{j=1}^{n} \frac{x_j^{\alpha_2} e^{-\hat{\lambda}_2^{old} x_j}}{f(x_j)}} \tag{3.8}$$

$$\hat{\alpha}_1^{new} = \frac{\ln\left(\frac{\psi(\hat{\alpha}_1)\sum_{j=1}^{n} \frac{e^{-\lambda_1 x_j} \lambda_1^{\hat{\alpha}_1^{old}} x_j^{\hat{\alpha}_1^{old} - 1}}{f(x_j)}}{\sum_{j=1}^{n} \frac{e^{-\lambda_1 x_j} x_j^{\hat{\alpha}_1^{old} - 1} \ln(\lambda_1 x_j)}{f(x_j)}}\right)}{\ln \lambda_1} \tag{3.9}$$

and
$$\hat{\alpha}_2^{new} = \frac{\ln\left(\dfrac{\psi(\hat{\alpha}_2)\sum_{j=1}^{n}\dfrac{e^{-\lambda_2 x_j}\lambda_2^{\hat{\alpha}_2^{old}} x_j^{\hat{\alpha}_2^{old}-1}}{f(x_j)}}{\sum_{j=1}^{n}\dfrac{e^{-\lambda_2 x_j} x_j^{\hat{\alpha}_2^{old}-1}\ln(\lambda_2 x_j)}{f(x_j)}}\right)}{\ln\lambda_2} \qquad (3.10)$$

where $\psi(x)$ is the digamma function which is defined as the logarithmic derivative of the gamma function or $\psi(x) = \frac{d}{dx}\log(\Gamma(x))$.

I set the following stopping rules for the gamma mixture model :

$$\begin{cases} |\hat{p}_1^{new} - \hat{p}_1^{old}| < 0.00001 \\ |\hat{\lambda}_1^{new} - \hat{\lambda}_1^{old}| < 0.005 \\ |\hat{\lambda}_2^{new} - \hat{\lambda}_2^{old}| < 0.005 \\ |\hat{\alpha}_1^{(new)} - \hat{\alpha}_1^{(old)}| < 0.005 \\ |\hat{\alpha}_2^{(new)} - \hat{\alpha}_2^{(old)}| < 0.005 \end{cases}$$

Output of the estimation will be shown in Table 3.4. However, we are not going to use these results directly either for the normal mixture model or for the gamma mixture model. In fact, these parameters will only be used in finding the suitable threshold point that allows us to differentiate the intermetallic region from the gold region.

Figure 3.6 to Figure 3.10 are five figures that show the estimated mixture model together with the original brightness density plot for five images produced by five different wire bonders. These five images are chosen to be the same images used in Figure 3.1 to Figure 3.5.

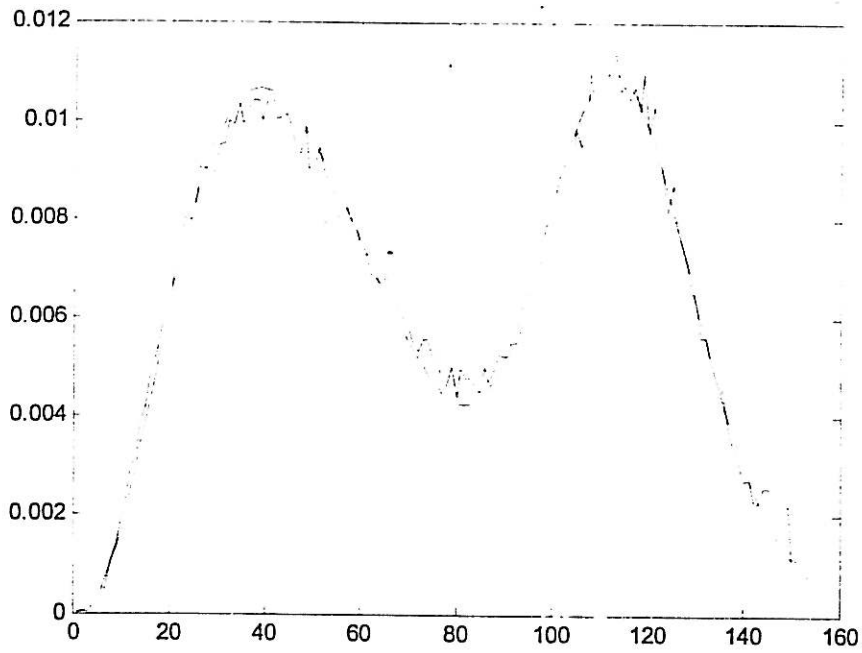


Figure 3.6 The line graph of brightness levels (blue) and the fitted gamma mixture model (green) for s101a.jpg

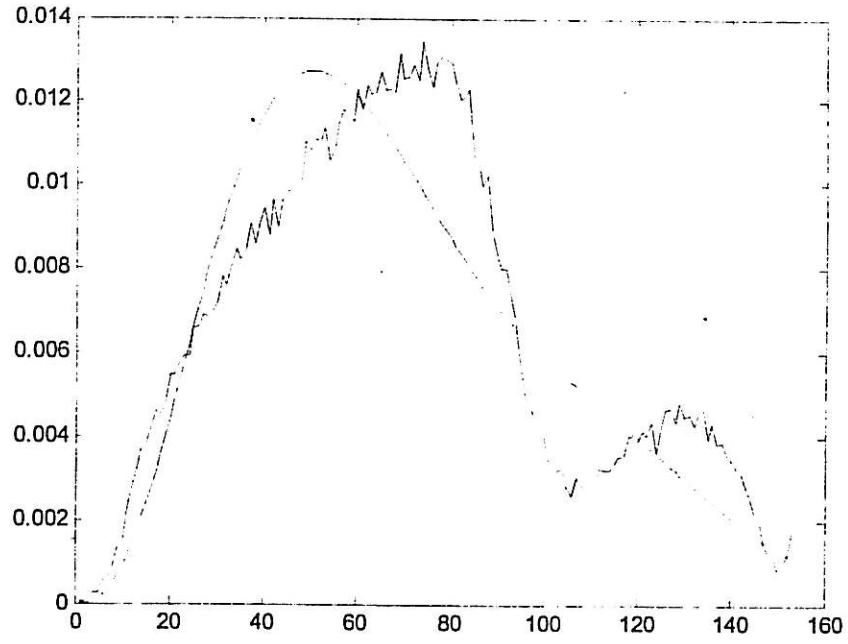


Figure 3.7 The line graph of brightness levels (blue) and the fitted gamma mixture model (green) for s201a.jpg

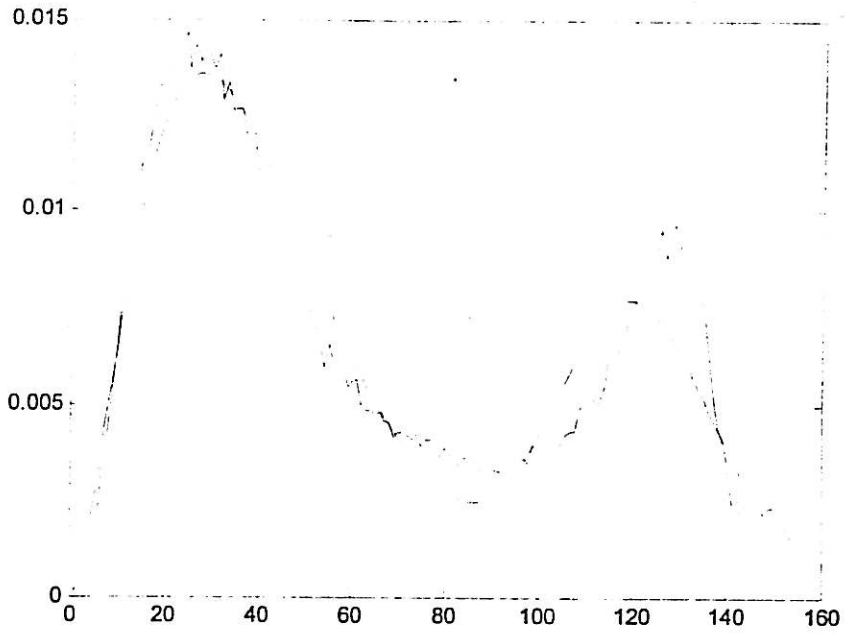


Figure 3.8 The line graph of brightness levels (blue) and the fitted gamma mixture model (green) for s301a.jpg

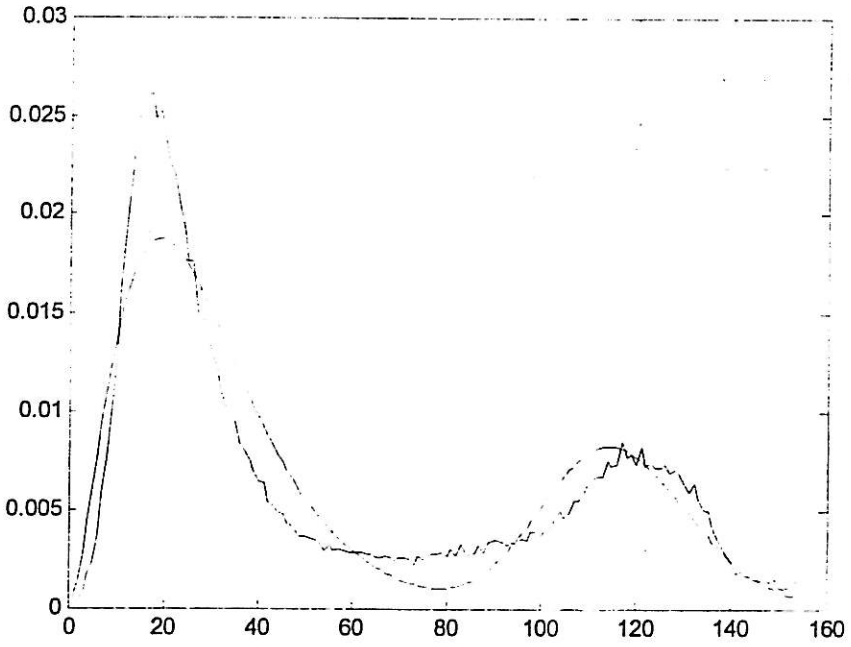


Figure 3.9 The line graph of brightness levels (blue) and the fitted gamma mixture model (green) for s401a.jpg

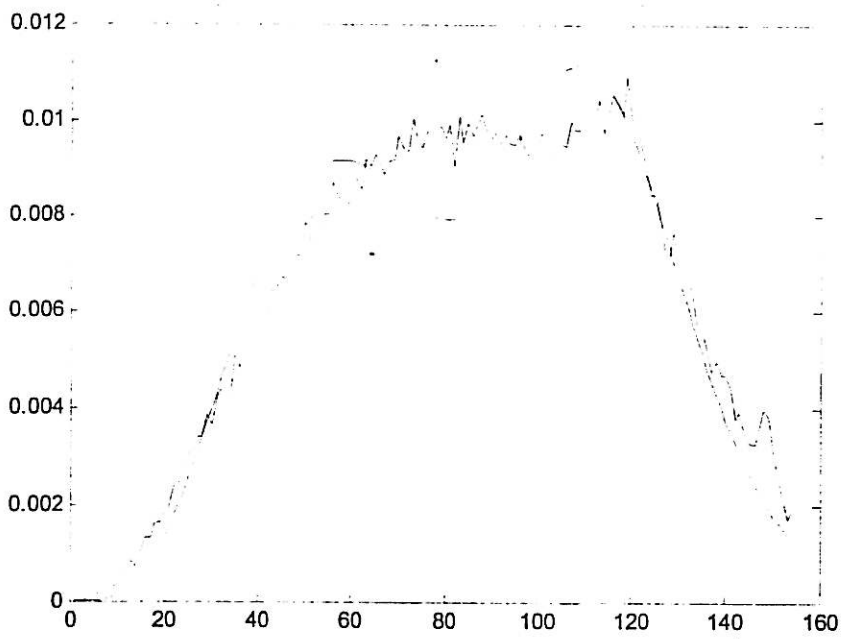


Figure 3.10 The line graph of brightness levels (blue) and the fitted gamma mixture model (green) for s501a.jpg

Similar to the case of fitting model of two-component normal mixture, we will use the Komogolov-Smirnov statistic to check our model. The results are shown in Table 3.5.

We can observe from the Komogolov-Smirnov statistic that, the gamma models seem to fit certain groups of wire bonding images better than the normal mixture models even though it still suggests rejection of the mixture model at significant level 0.01. For other groups of images, the model shows even bigger deviation from the brightness density compare to the two-component normal mixture models. The gamma estimates also take a longer time to converge.

No	Image	KS-stat	Critical Pt.
1	s101a	0. 00795	0. 00445
2	s102a	0. 01881	0. 00448
3	s103a	0. 01100	0. 00446
4	s104a	0. 01948	0. 00444
5	s105a	0. 00993	0. 00422
6	s106a	0. 01878	0. 00411
7	s201a	0. 04165	0. 00478
8	s202a	0. 02787	0. 00479
9	s203a	0. 04263	0. 00453
10	s204a	0. 04961	0. 00460
11	s205a	0. 03920	0. 00470
12	s206a	0. 03653	0. 00440
13	s301a	0. 01864	0. 00404
14	s302a	0. 03401	0. 00404
15	s303a	0. 01731	0. 00401

No	Image	KS-stat	Critical Pt.
16	s304a	0. 03398	0. 00403
17	s305a	0. 03809	0. 00407
18	s306a	0. 02115	0. 00405
19	s401a	0. 05531	0. 00409
20	s402a	0. 05092	0. 00395
21	s403a	0. 04601	0. 00400
22	s404a	0. 03829	0. 00408
23	s405a	0. 03488	0. 00415
24	s406a	0. 05748	0. 00393
25	s501a	0. 01956	0. 00417
26	s502a	0. 02743	0. 00427
27	s503a	0. 02927	0. 00417
28	s504a	0. 02804	0. 00423
29	s505a	0. 02596	0. 00433
30	s506a	0. 02268	0. 00406

Table 3.5 Comparison of Komogolov-Smirnov statistics for the gamma mixture model and corresponding critical points.

Chapter 4 Finding the Threshold Point and Edge Detection

In Chapter 3, we have fitted the model of the mixture of two normal distributions and of two gamma distributions to the brightness histograms. In this chapter we will use the model to estimate the percentage intermetallic and detect the boundary of the intermetallic region. The boundary is then used to calculate a measure called 'compactness' which will be used as an indicator of the uniformity of the intermetallic coverage. Two ways of thresholding, minimum expected cost threshold and minimum error threshold, will be studied in this project.

4.1 Minimum Expected Cost Threshold

In order to investigate the structure of the intermetallic coverage, we need to separate the intermetallic from the background, that is, the gold region. One way to achieve the separation is by setting a threshold point such that all pixels with brightness less than or equal to it are consider intermetallic while the pixels with brightness level more that the threshold are classified as gold. Hence, a binary image can be obtained with black representing the intermetallic region and white representing the gold region.

An immediate problem that we face is how to choose a suitable threshold point such that the misclassification probability is as small as possible.

Let π_1 and π_2 be two populations with respective probability density functions $f_1(x)$ and $f_2(x)$ as shown in Figure 4.1. Suppose we select T as the threshold point, then T divides the real line into two sets, those numbers less than or equal to T belong to the set R_1 while the set of numbers larger than T is denoted by R_2. The idea of thresholding is that when an observation falls in the region R_1 we will say that it belong to the population π_1. Similarly, when an observation falls in the region R_2 we will say that it belong to the population π_2. This way of classification inevitably leads to some misclassification error. However, we can

select a suitable threshold point such that the loss resulted from this misclassification is the least.

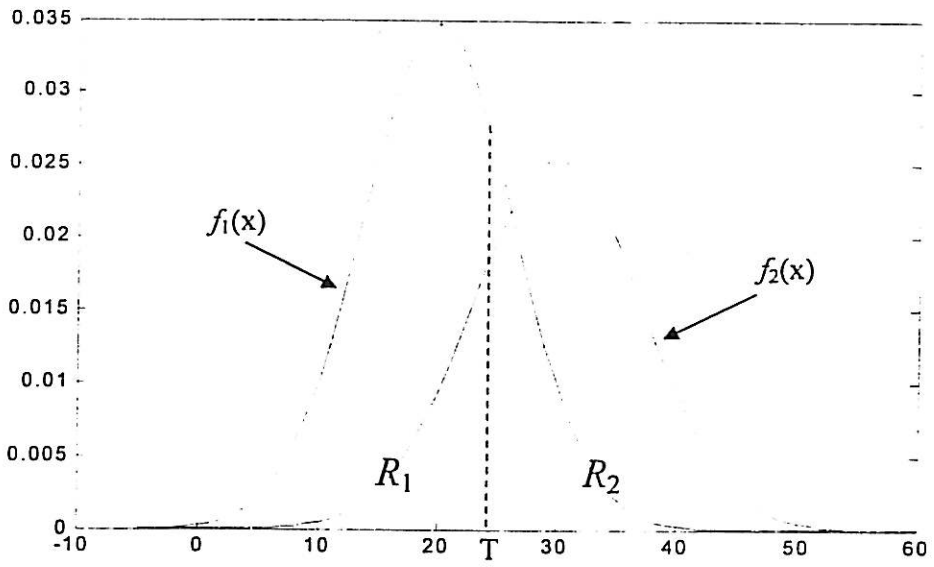


Figure 4.1 Illustration of the idea of thresholding

From conditional probability we have

$$P(x \in \pi_2 \text{ but is misclassified as } x \in \pi_1) = P(x \in R_1 | x \in \pi_2)\, p_2$$

$$P(x \in \pi_1 \text{ but is misclassified as } x \in \pi_2) = P(x \in R_2 | x \in \pi_1)\, p_1$$

where p_1 and p_2 are the prior probabilities of π_1 and π_2 respectively. If c(1|2) and c(2|1) represent the cost incurred when misclassify $x \in \pi_2$ as $x \in \pi_1$ and misclassify $x \in \pi_1$ as $x \in \pi_2$ respectively, the expected cost of misclassification (ECM) is given by

$$ECM = c(1|2)\, P(x \in R_1 | x \in \pi_2)\, p_2 + c(2|1)\, P(x \in R_2 | x \in \pi_1)\, p_1$$

In (Richard A. Johnson & Dean W. Wichern 2002) the region R_1 and R_2 that minimize the ECM are determined by the values of x for which the following inequalities hold:

$$x \in R_1 \text{ if } \frac{f_1(x)}{f_2(x)} \geq \left(\frac{c(1|2)}{c(2|1)}\right)\left(\frac{p_2}{p_1}\right) \text{ and } x \in R_2 \text{ if } \frac{f_1(x)}{f_2(x)} < \left(\frac{c(1|2)}{c(2|1)}\right)\left(\frac{p_2}{p_1}\right)$$

In this project, since there is no way to evaluate the cost incurred by misclassification we will assume c(1|2) = c(2|1), therefore the criteria reduces to

$$x \in R_1 \text{ if } \frac{f_1(x)}{f_2(x)} \geq \left(\frac{p_2}{p_1}\right) \text{ and } x \in R_2 \text{ if } \frac{f_1(x)}{f_2(x)} < \left(\frac{p_2}{p_1}\right)$$

This method can be easily applied to both the normal mixture model and the gamma mixture model. However, there is also situation when the method fails that is when only one of the inequalities described above hold. This happens when image s201a.jpg is fitted to the gamma mixture model. We can see from Figure 4.2 that $\frac{f_1(x)}{f_2(x)} \geq \left(\frac{p_2}{p_1}\right)$ for all x in the luminance range.

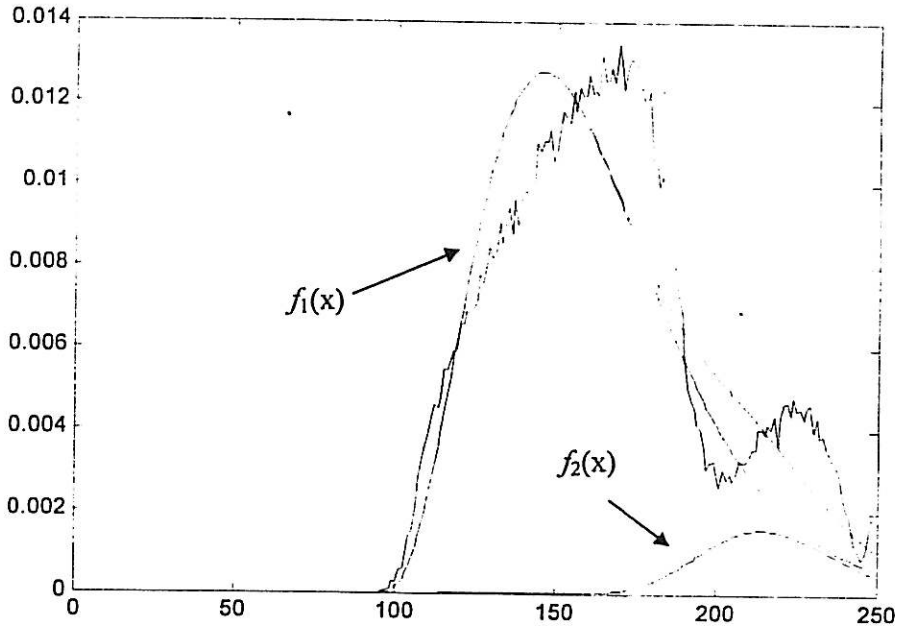


Figure 4.2 Gamma mixture model and component probability density functions $f_1(x)$ (red) and $f_2(x)$ (blue) for image s201a.jpg.

The resulting threshold points determined for both normal mixture model and gamma mixture models will be tabulated in Table 4.1.

Image	Bonder	Normal Mixture Threshold	%IM	Gamma Mixture Threshold	%IM
S101a	A	172	0.5191	185	0.5817
S102a	A	159	0.5836	177	0.6574
S103a	A	174	0.6141	186	0.6665
S104a	A	194	0.6364	194	0.6364
S105a	A	168	0.5765	185	0.6734
S106a	A	192	0.6516	194	0.6606
S201a	B	210	0.8700	NA	NA
S202a	B	185	0.8018	193	0.8204
S203a	B	199	0.8550	205	0.8683
S204a	B	203	0.8789	210	0.8934
S205a	B	194	0.8757	204	0.8960
S206a	B	176	0.8476	193	0.8892
S301a	C	168	0.6156	189	0.6915
S302a	C	155	0.6053	181	0.6889
S303a	C	164	0.6002	183	0.6608
S304a	C	147	0.5428	178	0.6369
S305a	C	146	0.5892	178	0.6819
S306a	C	166	0.6141	187	0.6886
S401a	D	142	0.5804	179	0.6866
S402a	D	145	0.5985	180	0.7074
S403a	D	146	0.5611	180	0.6646
S404a	D	149	0.6012	180	0.6877
S405a	D	150	0.5936	183	0.6797
S406a	D	139	0.5768	179	0.6911
S501a	E	190	0.5760	190	0.5760
S502a	E	187	0.5419	186	0.5299
S503a	E	189	0.6088	187	0.5851
S504a	E	191	0.5998	189	0.5809
S505a	E	180	0.6122	194	0.7410
S506a	E	189	0.6519	192	0.6750

Table 4.1 Threshold point and percentage intermetallic obtained by using minimum expected cost method on both normal mixture model and gamma mixture model.

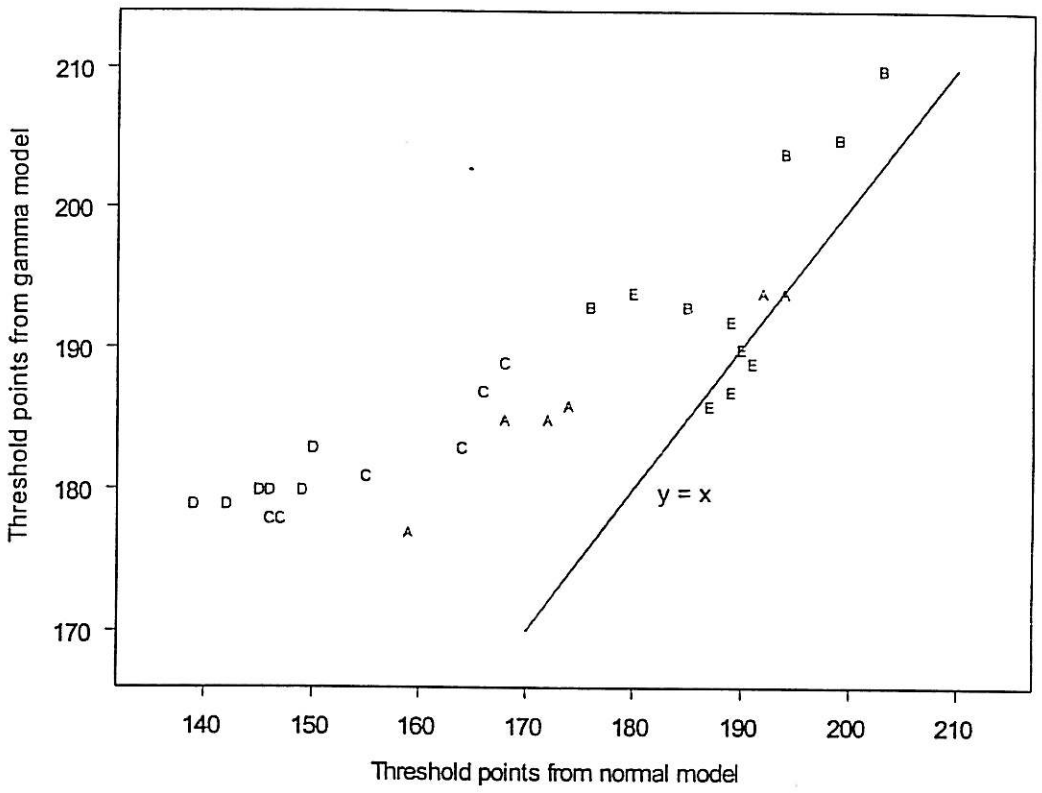


Figure 4.3 Scatter plot showing threshold points obtained from normal mixture model and gamma mixture model.

The wire bonds that we study are products of different machines. We will label the image according to the machine that produces them, namely, by using alphabet A, B, C, D and E. An immediate observation from Figure 4.3 is that the gamma mixture model gives larger value of threshold points for most of the images as compare to the normal mixture model. This could be caused by the positively skewed property of the gamma probability density function.

4.2 Minimum Error Threshold for the Normal Mixture Model

Another way of finding the threshold point is by considering the relative entropy between the fitted model and the original brightness histogram. The relative entropy (also known as Kullback-Leibler divergence) between two discrete probability distribution $p_1(x)$ and $p_2(x)$ is defined by

$$KL(p_1(x) \| p_2(x)) = \sum_i p_1(x_i) \log \frac{p_1(x_i)}{p_2(x_i)}$$

The relative entropy is a measurement of the discrepancy between two probability density functions. When the relative entropy is small, there is less discrepancy. We can thus compare the discrepancy between the brightness histogram and the mixture model. The idea of minimum error threshold is to find a threshold point such that the relative entropy between the fitted model and the brightness histogram is the smallest. That is equivalent to finding a point such that the brightness histogram is closest to the fitted model.

The following criterion was derived in the work of Fan Jiulun & Xie Winxin (1997). I will use the result without repeating the derivation.

First of all, if we divide the brightness histogram into two parts by selecting an arbitrary brightness level t and modeling each part with a normal distribution and comparing the model to the histogram by means of relative entropy. The smaller is the relative entropy, the larger is the match between the model and the histogram.

In (Fan Jiulun *et al.*, 1997) it was shown that this is equivalent to finding t that minimize

$$J(t) = [P_1(t) \ln \sigma_1(t) + P_2(t) \ln \sigma_2(t)] - [P_1(t) \ln P_1(t) + P_2(t) \ln P_2(t)],$$

where $$P_1(t) = \sum_{g=0}^{t} h(g), \quad P_2(t) = \sum_{g=t+1}^{T} h(g),$$

$$\mu_1(t) = \frac{\sum_{g=0}^{t} h(g)g}{P_1(t)}, \qquad \mu_2(t) = \frac{\sum_{g=t+1}^{T} h(g)g}{P_2(t)},$$

$$\sigma_1^{\,2}(t) = \frac{\sum_{g=0}^{t} (g-\mu_1(t))^2 h(g)}{P_1(t)} \quad \text{and} \quad \sigma_2^{\,2}(t) = \frac{\sum_{g=t+1}^{T} (g-\mu_2(t))^2 h(g)}{P_2(t)}.$$

The T in the equation above is the highest brightness level in the data and $h(g)$ is the frequency of brightness level g in the brightness histogram. A big advantage of the minimum error threshold method is that we don't have to find the component in the mixture model to carry out the thresholding. Hence, the whole process in chapter 3 can be skipped. Table 4.2 shows the comparison of the threshold points obtained by using the minimum expected cost method (MEC) and that by using minimum error method (ME) for the normal mixture model.

No	Image	MEC	ME
1	s101a	172	172
2	s102a	159	158
3	s103a	174	172
4	s104a	194	190
5	s105a	168	168
6	s106a	192	186
7	s201a	210	198
8	s202a	185	179
9	s203a	199	190
10	s204a	203	195
11	s205a	194	184
12	s206a	176	169
13	s301a	168	164
14	s302a	155	154
15	s303a	164	162

No	Image	MEC	ME
16	s304a	147	147
17	s305a	146	146
18	s306a	166	163
19	s401a	142	142
20	s402a	145	145
21	s403a	146	146
22	s404a	149	149
23	s405a	150	149
24	s406a	139	139
25	s501a	190	183
26	s502a	187	179
27	s503a	189	178
28	s504a	191	182
29	s505a	180	177
30	s506a	189	179

Table 4.2 Comparison of the threshold points obtained by the minimum expected cost method and minimum error method on the normal mixture model.

From Table 4.2, we can see that the minimum expected cost method always gives higher or equal to threshold point compare to the minimum error method. This also implies that the percentage intermetallic obtained by using the minimum expected cost method is always higher or equal to that obtained by minimum error method.

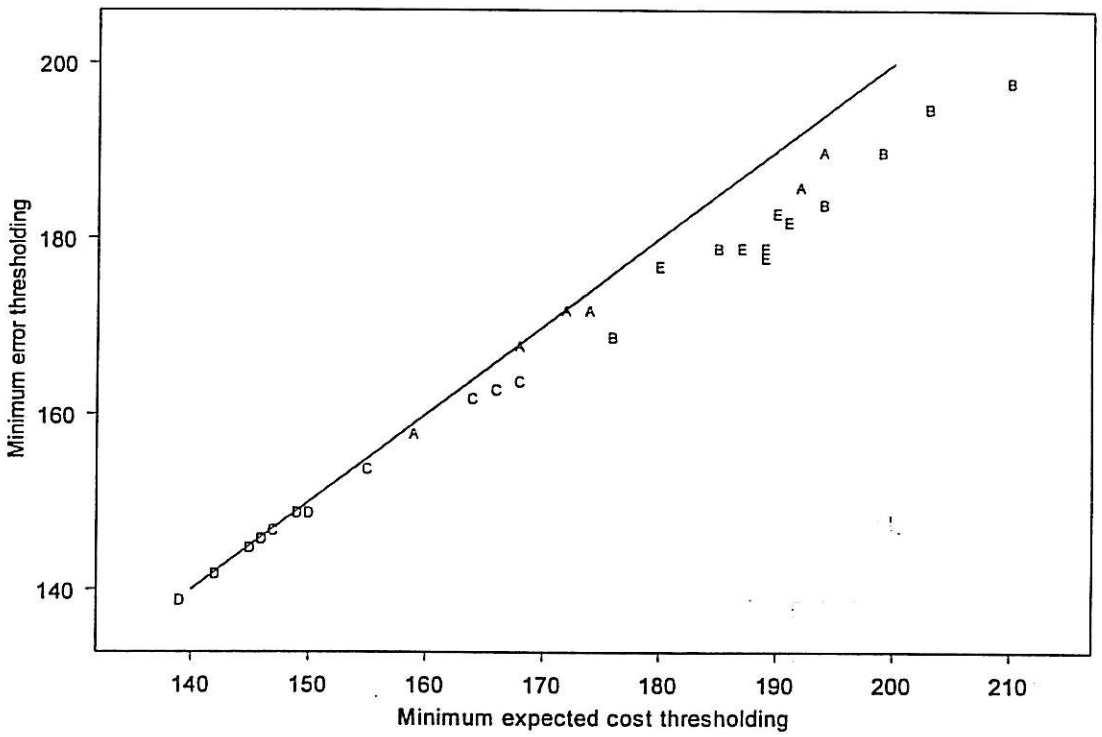


Figure 4.4 Comparison of the threshold points estimated by the minimum expected cost method and minimum error method base on the normal mixture model.

Using a thresholds point, we can set all those pixels with brightness level less than it 0 while all pixels with brightness level higher or equal to it 1. Thus getting a binary image, denoted by BI. The ratio of zeros to the total number of pixels in the binary image BI gives the percentage intermetallic of the wire bond. The percentage intermetallic estimated by the minimum expected cost method and minimum error method is shown in Table 4.3.

No	Image	MEC	ME
1	s101a	0. 5191	0. 5191
2	s102a	0. 5836	0. 5797
3	s103a	0. 6141	0. 6048
4	s104a	0. 6364	0. 6218
5	s105a	0. 5765	0. 5765
6	s106a	0. 6516	0. 6229
7	s201a	0. 8700	0. 8337
8	s202a	0. 8018	0. 7868
9	s203a	0. 8550	0. 8354
10	s204a	0. 8789	0. 8640
11	s205a	0. 8757	0. 8539
12	s206a	0. 8476	0. 8247
13	s301a	0. 6156	0. 5991
14	s302a	0. 6053	0. 6015
15	s303a	0. 6002	0. 5934

No	Image	MEC	ME
16	s304a	0. 5428	0. 5428
17	s305a	0. 5892	0. 5892
18	s306a	0. 6141	0. 6014
19	s401a	0. 5804	0. 5804
20	s402a	0. 5985	0. 5985
21	s403a	0. 5611	0. 5611
22	s404a	0. 6012	0. 6012
23	s405a	0. 5936	0. 5909
24	s406a	0. 5768	0. 5768
25	s501a	0. 5760	0. 5088
26	s502a	0. 5419	0. 4483
27	s503a	0. 6088	0. 4767
28	s504a	0. 5998	0. 5150
29	s505a	0. 6122	0. 5809
30	s506a	0. 6519	0. 5715

Table 4.3 Comparison of the percentage intermetallic estimated by the minimum expected cost method and minimum error method on the normal mixture model.

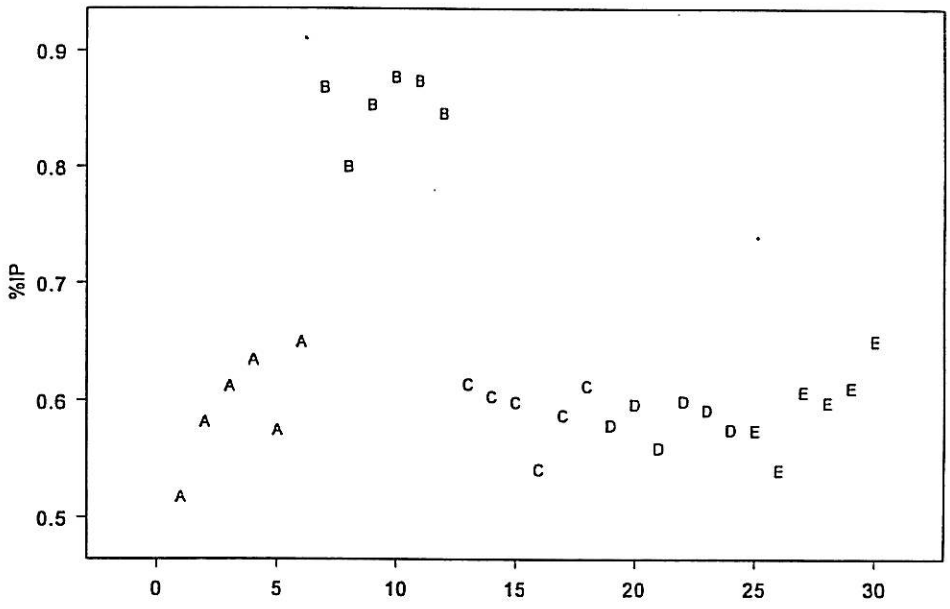


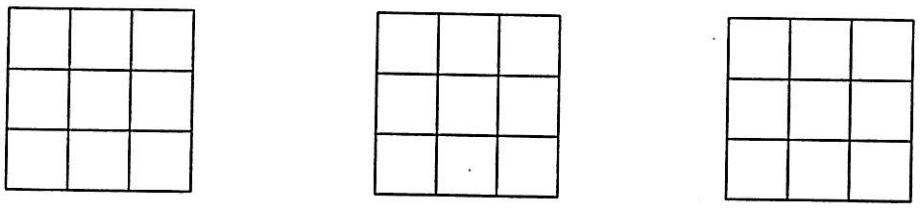
Figure 4.5 Scatter plot showing the percentage intermetallic of bonds produced by different wire bonders based on MEC thresholding.

The scatter plots in Figures 4.5 give us a better view of the results shown in Table 4.3. It shows that the percentage intermetallic of the wire bond produced by the same machine are quite consistent, except those produced by bonder labeled B.

4.3 Edge detection

The threshold points obtained in section 4.2 is very useful to us in the sense that it provides an objective way for us to separate the gold from the intermetallic compound.

There are many ways of defining the edge of a region. For our purpose, I will define the edge of an intermetallic region as the set of pixels that have at least one different values with its neighborhood in the binary image BI. The neighborhood of a pixel (i, j) can be defined in several ways. Figure 4.6 illustrates three of the commonly used definition. In Figure 4.6, the shaded pixels can be regarded as the neighborhood of the pixel at the center. We are going to use the definition of neighborhood shown in Figure 4.6 (c) to avoid repeated counting in the algorithms.



(a) (b) (c)

Figure 4.6 Different ways of defining the neighborhood of a pixel

A pixel is said to be a point on the edge when it has different pixel values with any one of the pixels in its neighborhood. From the binary image BI, we proceed by searching the pixels one by one starting from the upper left corner to the lower right.

Let BI(x, y) represent the brightness level of the (x, y) pixel which is either 1 or 0 in the image BI. We will create a new matrix E whose value at the (i, j) location is 0 if and only if a pixel located at (i, j) in BI satisfy

$$| 2*BI(i, j) - BI(i+1, j) - BI(i, j+1)) | > 0.$$

The condition above is actually an alternative way to say if any of the values of the pixel BI(i+1, j) or BI(i, j+1) is/are not the same as the value of pixel BI(i, j) then E(i, j) will be marked as boundary. All others pixels will be given value 1.

From the image BI and E that we obtained, we can count the area and perimeter of the intermetallic region. The number of 0 in the BI image will be regarded as the area of the intermetallic region while the number of 0 in the E image will be treated as the length of the perimeter. This idea is justify by imagine each pixel is a small square with area one unit and side with length 1 unit. A binary image and its edge obtained by using the methods mentioned above is shown in Figure 4.7.

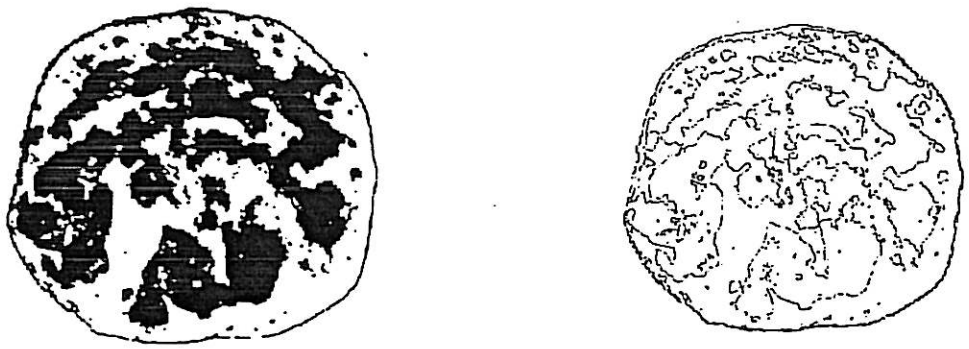


Figure 4.7 Binary image and image of boundary for image s101a.jpg by using minimum error thresholding on normal mixture model.

4.4 Compactness

In pattern recognition, compactness is one of the indicators used to described the feature of an image. Compactness of a region is defined by $\frac{perimeter^2}{4\pi \times area}$. The formula can also be written in the form $\frac{\frac{perimeter^2}{4\pi}}{area}$ where the numerator is the area of circle that can be bounded by a particular perimeter. The compactness is thus represent the ratio of the area of a circle bounded by a particular perimeter and the area of the shape bounded by the same perimeter. It reflects the efficiency in using the perimeter in bounding an object. (Mark Nixon & A. Aguado, 2002)

A perfect circle has compactness equal to 1, the compactness become bigger and bigger when the image has curly and twisted boundary. Therefore by find the compactness of an image we can measure the degree of 'twist' and uniformity of an intermetallic coverage.

The idea of compactness as a measurement of the uniformity of a region can be illustrated by the following figures. I will compare the compactness of different images to artificial diagram with circular or elongated spots. This can give us an idea of the discrepancy of the intermetallic coverage in our image from the circle.

Figure 4.8 shows two artificial figure with their respective compactness calculated by using the total area of the black regions and the total boundary length of the regions.

Figure 4.8 a) Compactness = 4.7 b) Compactness= 17.6

In Figure 4.7, the compactness measure of two wire bond images is calculated. It is obvious that the image with densely distributed and higher percentage intermetallic produce smaller compactness whereas the one with lower percentage intermetllic and scattered intermetallic coverage gives much higher compactness measure.

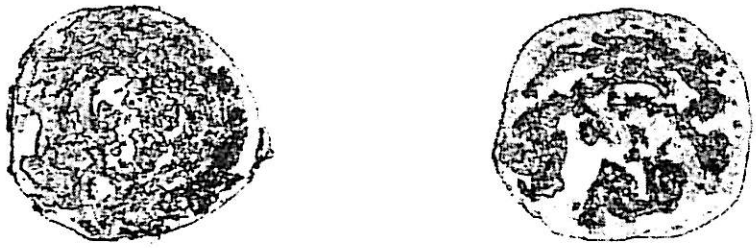


Figure 4.9 a) S201a.jpg, Compactness = 50 b) S101a.jpg, Compactness = 150

As we have mentioned in Chapter 1, both percentage intermetallic and uniformity are crucial in determining the quality of a wire bond. We will thus investigate the possible relationship between the two variables. Table 4.4 is the tabulated results of percentage intermetallic and compactness. Correlation coefficients grouped by machine are calculated in Table 4.5. We can see that there are some relationships between two variables. Figure 4.10 gives an even better picture by showing the scatter plot of the two variables.

No	Image	%IM	Compactness
1	s101a	0. 5191	150
2	s102a	0. 5836	128
3	s103a	0. 6141	128
4	s104a	0. 6364	94
5	s105a	0. 5765	170
6	s106a	0. 6516	128
7	s201a	0. 8700	39
8	s202a	0. 8018	40
9	s203a	0. 8550	33
10	s204a	0. 8789	27
11	s205a	0. 8757	28
12	s206a	0. 8476	50
13	s301a	0. 6156	104
14	s302a	0. 6053	102
15	s303a	0. 6002	88

No	Image	%IM	Compactness
16	s304a	0. 5428	94
17	s305a	0. 5892	108
18	s306a	0. 6141	97
19	s401a	0. 5804	84
20	s402a	0. 5985	123
21	s403a	0. 5611	96
22	s404a	0. 6012	83
23	s405a	0. 5936	88
24	s406a	0. 5768	86
25	s501a	0. 5760	341
26	s502a	0. 5419	425
27	s503a	0. 6088	288
28	s504a	0. 5998	262
29	s505a	0. 6122	252
30	s506a	0. 6519	172

Table 4.4 Percentage intermetallic and compactness of wire bonds by minimum expected cost thresholding.

Machine	Correlation coefficient between percentage intermetallic and compactness
A	-0.6382
B	-0.5172
C	0.2671
D	0.1892
E	-0.9842

Table 4.5 Correlation coefficients between percentage intermetallic and compactness as grouped by machine.

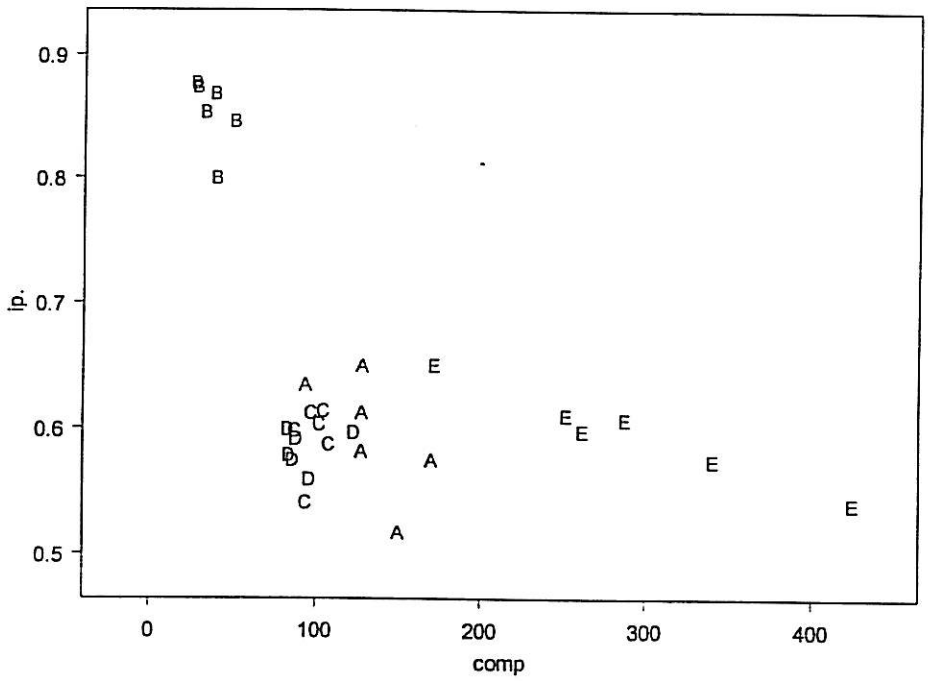


Figure 4.10 Scatter plot of percentage intermetallic versus compactness.

The scatter plot above may suggest that when the percentage intermetallic is high, the compactness remains small. However, when percentage intermetallic drops below certain level, the fluctuations can be quite big. It implies that when the percentage intermetallic is low, there can be bigger chance to get a badly bonded wire.

Chapter 5 Conclusion and Discussions

Due to the restriction of physical environment, I do not have the privilege to work in the manufacturing plant to carry out the experiment myself. This project is based on 30 existing images provided by a semiconductor manufacturer. Unfortunately, the bonding parameters used in producing them are no longer available. Therefore, it is difficult to verify some of the results obtained, especially those that involve the quality of wire bond.

In the industry, I would suggest that this procedure to be carried out to evaluate the quality of a bonding machine. The shear test alone might not be sufficient to evaluate the long run quality of a wire bond. This is because if the intermetallic coverage is not uniform the voids can easily join together at high temperature and hence resulting cracks. As we can see from the results in Chapter 4, the graphs reveals the two most important characteristics of a good bond quite effectively. This method can also be used to evaluate the performance of a set of bonding parameters.

We have also seen that the fitted model is not the true model as revealed by the Komogolov-Simirnov test. We can only treat it as an approximation to the true model. A better model could be found if we investigate the mechanism of the JPEG image structure and get a better idea on how such interference of colours happens. An even more straightforward way is that we can use the Bitmap file that preserve more original information instead of using the JPEG file that use compression techniques when storing the images.

I have included two methods of thresholding in the project. Question remained unanswered on which of these described the truth the best. Further experiments need to be carried out to verify the results.

For the completion of the project, I have also prepared another set of images that are obtained from the original colour images by removing the outer boundary of the wire bond that is brown in colour. This region is not part of the intermetallic region. However, the thresholding technique used will classify it into intermetallic compound due to its dark colour. Although this phenomenon has small effect on the intermetallic percentage estimated, it affects the compactness of the image drastically. The boundaries were removed by using the 'rubber' tool in the Microsoft Paint. Although the calculated compactness of these set of 'polished' images reduce significantly to smaller values, the pattern of the distributions does not change significantly. The output will be shown in Appendix D.

Bibliography

[1] Chris Otter, *Wire bonding in microelectronics*, TWI Ltd. http://www.twi.co.uk/j32k/protected/band_3/kscco001.html

[2] *The Nordic Electronics Packaging Guideline.* (2000) Danish Electronics, Lights & Acoustics; The Swedish Institute of Production Engineering Research; SINTEF, Norway; Technical Research Center of Finland. (http://extra.ivf.se/ngl/)

[3] Everitt, B.S. & Hand, D.J. (1981). *Finite Mixture Distributions*. London: Chapman and Hall Ltd.

[4] Fan Jiulun & Xie Winxin (1997), *Minimum Error Thresholding: A note.* Patttern Recognition Letters 18 (1997), pg705 – pg709

[5] Gazali Bin Omar (1996), *Reliability of Wire Bonding in Microelectronic Packaging,* Master Thesis, University of Malaya.

[6] J. Murdoch & J. A. Barnes (1998), *Statistical Tables 4th Edition*, Palgrave MacMillan.

[7] John C. Russ (1995), The Image Processing Handbook 2nd Edition, CRC Press, Inc. USA.

[8] J. V. Deshpande & U. V. Naik-Nimbalkar, *Testing Goodness of Fit of a Specific Parametric Probability Model.* Uncertainty and Optimality: Probability, Statistics and Operations Research, World Scientific Publishing Co. Pte. Ltd. (2002)

[9] Keiran Foster, *Wire Bonding Techniques*, Central Electronics Group http://www.physics.ox.ac.uk/electronics/Default.htm

[10] Ker-Chang, Hsieh (2002), 电子构装微接点的高温可靠度研究 *Reliability of the Wire Bond Interface at High Temperature*, Research Project No: NSC 89-2216-E-110-040, National Science Council, Taiwan.

[11] Mark Nixon & Alberto Aguado (2002), Feature Extraction and Image Processing, Newnes, United Kingdom.

[12] Norliza.M. Noor, O.M. Rijal, O.M. Badar & Y.T. P'ng (2000), *A Vision Problem In Wire Bonding*, The Proceedings of the IEEE TENCON 2000, Kuala Lumpur, pp.1299-1304.

[13] Norliza.M. Noor, O.M. Rijal, O.M. Badar & Y.T. P'ng (2001), *A Mixture Distribution for the Wire Bonding Image*, International Symposium on Signal Processing and its Applications, Kuala Lumpur, Malaysia.

[14] Omar M. Rijal, Norliza M. Noor, O. M. Badar and Y.T. P'ng (1999), "Digital Image Analysis For Measuring Intermetallics Formation", Technical Report, Assembly Technical Department, Texas Instruments Malaysia.

[15] Philofsky, E., *Intermetallic Formation in Gold-Aluminum Systems*, Solid State Electronics, Vol. 13, 1970, 1391-1300.

[16] Richard A. Johnson & Dean W. Wichern (2002), Applied Multivariate Statistical Analysis, Fifth Edition, Pearson Education International.

[17] Servais,G.E. & Brandenburg, S.D.(1991), *Wire Bonding – A Closer Look*, ISTFA'91, Los Angeles, USA.

[18] Suresh Kumar, Frank Wulff & Klaus Dittmer, *Degradation of Small Ball Bonds due to Intermetallic Phase(IP) Growth*, Kulicke & Soffa. http://www.kns.com/resources/articles/kumar.pdf

[19] William B Pennebaker & Loan L. Mitchell (1993), *JPEG: Still Image Data Compression Standard,* Van Nostrand Reinhold, New York, USA.

Appendix A

The Bhattacharya Graphical Methods

The Bhattacharya graphical method for grouped data is suggested by Bhattacharya in 1967. It is used to find the initial estimates of parameters in the model of a finite mixture distribution.

To implement the technique we need to group the data in the first step. In our case, our data is the brightness histogram generated from a grey scale image. However, the brightness histogram is not directly usable for this method because the number of groups is too big since every brightness level corresponds to one group. Therefore, grouping the data meaning combining several brightness levels in one group.

We first capture the part of the histograms that contains information about the image, that is from brightness level $\frac{101}{255}$ to $\frac{250}{255}$ truncating both ends that contain either noises or blank. In this project the data is grouped into 15 classes with boundaries $\frac{101}{255}$ to less than $\frac{110}{255}$, $\frac{110}{255}$ to less than $\frac{120}{255}$ and so on. The frequency of each group will be stored in a matrix. The following is the output of the grouped data using image s101a.jpg as example.

[2887	8899	12699	13622	12091	9851	7737	6333
7145	11010	14042	13067	8295	3889	2257]	

The grouped frequency will then be used to calculate $\log\frac{\phi_{i+1}}{\phi_i}$, where ϕ_i is the frequency of the i^{th} class, for i = 1 to 14. For example in the output above ϕ_1 is 2887 and we will store the results in another matrix, which contains only 14 elements.

[1.1257 0.3556 0.0702 -0.1192 -0.2049 -0.2416 -0.2002
0.1206 0.4324 0.2432 -0.0720 -0.4544 -0.7575 -0.5441]

$\log\frac{\phi_{i+1}}{\phi_i}$ are then plotted against the midpoints of the classes. The scatter plot is shown in the following diagram.

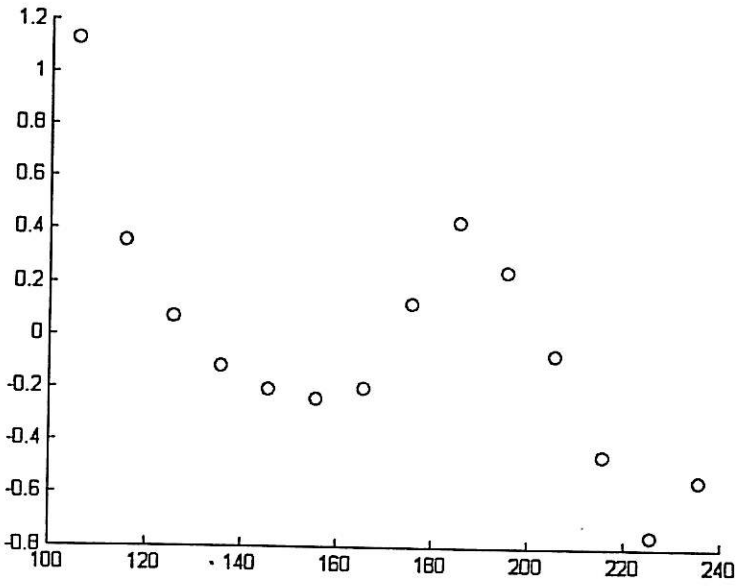


Figure A.1 Scatter plot of $\log\frac{\phi_{i+1}}{\phi_i}$ against the midpoints.

A geometrical interpretation of $\log\frac{\phi_{i+1}}{\phi_i}$ is that, it is positive when the $(i+1)^{th}$ class has higher frequency than the i^{th} class and negative if the $(i+1)^{th}$ class has lower frequency than the i^{th} class. Therefore, when $\log\frac{\phi_{i+1}}{\phi_i}$ change from positive to negative it represents a hump in the brightness histogram. From the scatter plot we can see that these points can be fitted to two lines with negative slopes which correspond to the two humps in the brightness histogram. The slope and x-intercept of the lines are calculated and will be used to find the initial estimates. The fitted

linear regression lines were found by using Splus. The output and the fitted equations are shown in Figure A.2.

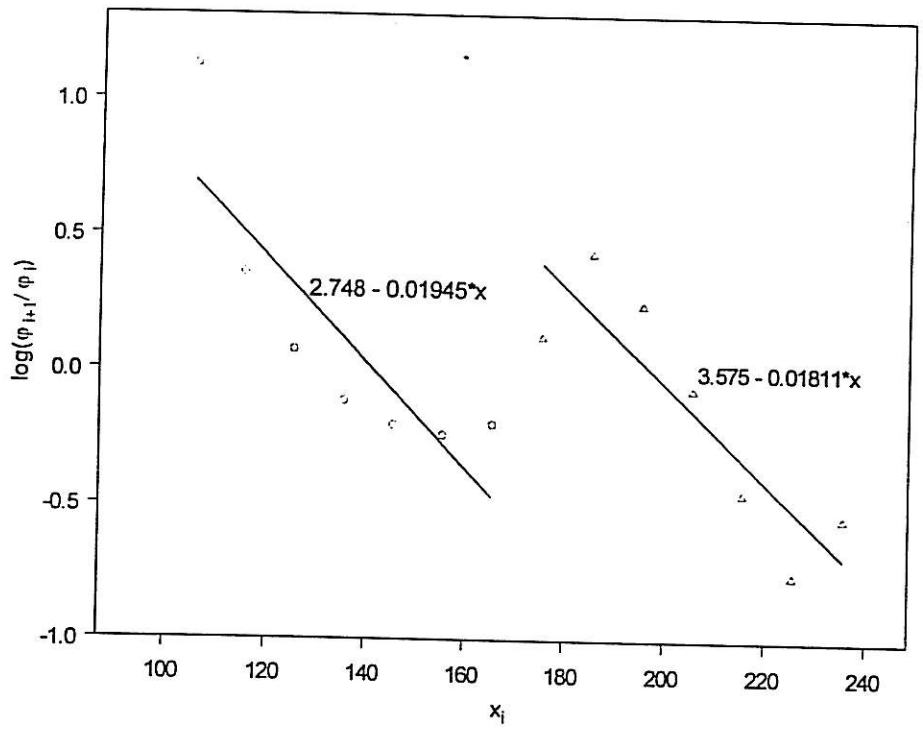


Figure A.2 Linear regression lines fitted to the scatter plot of $\log\frac{\phi_{i+1}}{\phi_i}$ against the midpoints.

The estimates of the parameter μ_k and σ_k^2 of the k[th] component can be calculated by

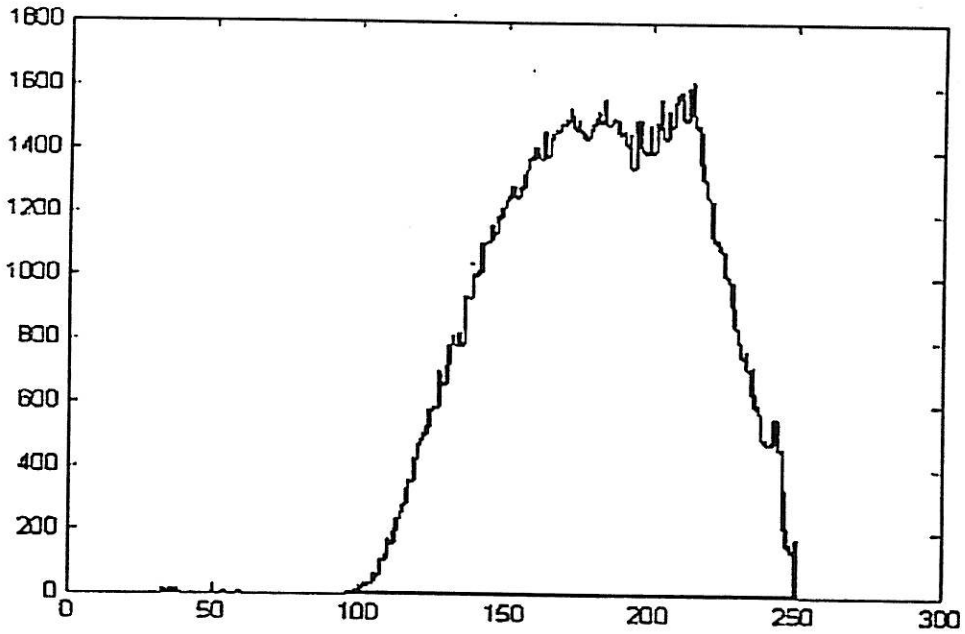
$$\hat{\mu}_k = \lambda_k + w/2$$

$$\text{and } \sigma_k^2 = w\cot(\alpha_k) - \frac{w^2}{12}$$

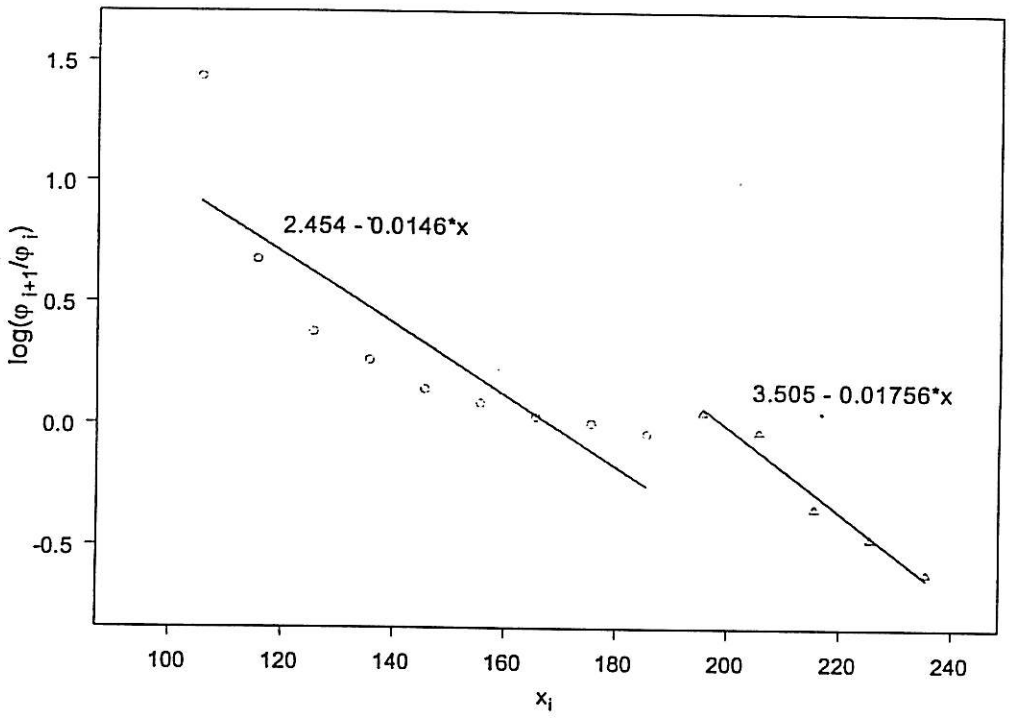
where α_k is the angle between the k[th] straight line and the negative direction of the x-axis, λ_k is the x-intercept of the line and w is the class width which is 10 in this case.

From the first fitted line, $\lambda_1 = \frac{2.748}{0.01945} = 141.285$, $\cot\alpha_1 = \frac{1}{0.01945} = 51.413$, hence we get $\hat{\mu}_1 = 146.285$ and $\hat{\sigma}_1^2 = 505.797$. Similarly, we have $\hat{\mu}_2 = 202.40$ and $\hat{\sigma}_2^2 = 543.87$. The results can then be used in the EM algorithm to find more accurate estimates of the parameters.

For a histogram with well-separated humps, the initial estimates obtained by the Bhattacharya method works tends to lessen the number of iterations needed for convergence in the EM algorithm. However, for the brightness histogram like that of image s501a.jpg which is shown in Figure A.3 (a), we will get the scatter plot in Figure A.3 (b), where the graph is not obviously fitted by two lines. The graphical estimates tends to increase the number of iterations needed in this case.



(a)



(b)

Figure A.3 (a) Brightness histogram for image s501a.jpg and (b) linear regression lines fitted to the scatter plot of $\log\frac{\phi_{i+1}}{\phi_i}$ against the midpoints.

Appendix B

Derivation of MLE for the normal mixture model

The probability density function of the mixture of two normal distributions $N(\mu_1,\sigma_1^2)$ and $N(\mu_2,\sigma_2^2)$ in proportion of p_1 and $1-p_1$ respectively can be written explicitly as $f(x)=\frac{p_1}{\sqrt{2\pi\sigma_1^2}}\exp\left(-\frac{1}{2}\frac{(x-\mu_1)^2}{\sigma_1^2}\right)+\frac{1-p_1}{\sqrt{2\pi\sigma_2^2}}\exp\left(-\frac{1}{2}\frac{(x-\mu_2)^2}{\sigma_2^2}\right)$.

In (1.2), we have seen that

$$\hat{p}_i^{new}=\frac{1}{n}\sum_{j=1}^{n}\frac{\hat{p}_i^{old}g_i(x_j)}{f(x_j)}$$

is a maximum likelihood estimator for the proportion parameter p_1.

By taking $$g_1(x)=\frac{1}{\sqrt{2\pi\sigma_1^2}}\exp\left(-\frac{1}{2}\frac{(x-\mu_1)^2}{\sigma_1^2}\right)$$

we get

$$\hat{p}_1^{new}=\frac{1}{n}\sum_{j=1}^{n}\frac{\frac{\hat{p}_1^{old}}{\sqrt{2\pi\sigma_1^2}}\exp\left(-\frac{1}{2}\frac{(x_j-\mu_1)^2}{\sigma_1^2}\right)}{f(x_j)}$$

as the estimator for the first proportion parameter.

From (1.3), we have also seen that the maximum likelihood estimators of parameter in $g_1(x)$ and $g_2(x)$ must satisfy

$$\sum_{j=1}^{n}\frac{p_q\frac{\partial}{\partial\theta_{qm}}g_q(x_j)}{f(x_j)}=0$$

Suppose we want to find an estimator for μ_1, put $\theta_{qm} = \mu_1$ and we require

$$\sum_{j=1}^{n} \frac{p_1 \frac{\partial}{\partial \mu_1}\left(\frac{1}{\sqrt{2\pi\sigma_1^{\,2}}}\exp\left(-\frac{1}{2}\frac{(x_j-\mu_1)^2}{\sigma_1^{\,2}}\right)\right)}{f(x_j)} = 0$$

$$\sum_{j=1}^{n} \frac{\frac{p_1}{\sqrt{2\pi\sigma_1^{\,2}}}\exp\left(-\frac{1}{2}\frac{(x_j-\mu_1)^2}{\sigma_1^{\,2}}\right)\left(\frac{x_j-\mu_1}{\sigma_1^{\,2}}\right)}{f(x_j)} = 0$$

Dividing $\frac{p_1}{\sigma_1^{\,2}}$ on both sides we can get

$$\sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j-\mu_1)^2}{\sigma_1^{\,2}}\right)(x_j-\mu_1)}{f(x_j)} = 0$$

$$\sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j-\mu_1)^2}{\sigma_1^{\,2}}\right)x_j}{f(x_j)} = \sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j-\mu_1)^2}{\sigma_1^{\,2}}\right)\mu_1}{f(x_j)}$$

$$= \mu_1 \sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j-\mu_1)^2}{\sigma_1^{\,2}}\right)}{f(x_j)}$$

Therefore, we may estimate μ_1 by

$$\hat{\mu}_1^{\,new} = \frac{\sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j-\hat{\mu}_1^{\,old})^2}{\sigma_1^{\,2}}\right)x_j}{f(x_j)}}{\sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j-\hat{\mu}_1^{\,old})^2}{\sigma_1^{\,2}}\right)}{f(x_j)}}$$

Similarly, we can get $$\hat{\mu}_2^{new} = \frac{\sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j - \hat{\mu}_2^{old})^2}{\sigma_1^2}\right) x_j}{f(x_j)}}{\sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j - \hat{\mu}_2^{old})^2}{\sigma_1^2}\right)}{f(x_j)}}$$

For the estimation of σ_1^2 we require,

$$\sum_{j=1}^{n} \frac{p_1 \frac{\partial}{\partial \sigma_1^2}\left(\frac{1}{\sqrt{2\pi\sigma_1^2}} \exp\left(-\frac{1}{2}\frac{(x_j - \mu_1)^2}{\sigma_1^2}\right)\right)}{f(x_j)} = 0$$

$$\sum_{j=1}^{n} \frac{p_1\left(\frac{1}{2\sqrt{2\pi\sigma_1^2}} \exp\left(-\frac{1}{2}\frac{(x_j - \mu_1)^2}{\sigma_1^2}\right)\left(\frac{x_j - \mu_1}{\sigma_1^2}\right)^2 - \frac{1}{2\sqrt{2\pi}\sigma_1^3} \exp\left(-\frac{1}{2}\frac{(x_j - \mu_1)^2}{\sigma_1^2}\right)\right)}{f(x_j)} = 0$$

Dividing $\frac{p_1}{2\sqrt{2\pi}\sigma_1^3}$ on both sides and extracting common factor, we have

$$\sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j - \mu_1)^2}{\sigma_1^2}\right)\left(\frac{(x_j - \mu_1)^2}{\sigma_1^2} - 1\right)}{f(x_j)} = 0$$

$$\frac{1}{\sigma_1^2}\sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j - \mu_1)^2}{\sigma_1^2}\right)(x_j - \mu_1)^2}{f(x_j)} = \sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j - \mu_1)^2}{\sigma_1^2}\right)}{f(x_j)}$$

Finally, we have $$\hat{\sigma}_1^{2(new)} = \frac{\sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j - \mu_1)^2}{\hat{\sigma}_1^{2(old)}}\right)(x_j - \mu_1)^2}{f(x_j)}}{\sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j - \mu_1)^2}{\hat{\sigma}_1^{2(old)}}\right)}{f(x_j)}}$$

Similarly,

$$\hat{\sigma}_2^{2(new)} = \frac{\sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j-\mu_2)^2}{\hat{\sigma}_2^{2(old)}}\right)(x_j-\mu_2)^2}{f(x_j)}}{\sum_{j=1}^{n} \frac{\exp\left(-\frac{1}{2}\frac{(x_j-\mu_2)^2}{\hat{\sigma}_2^{2(old)}}\right)}{f(x_j)}}$$

Appendix C

Derivation of MLE for gamma mixture model

The probability density function of the mixture model of two gamma distributions $g_1(x) = \frac{\lambda_1^{\alpha_1} x^{\alpha_1 - 1} e^{-\lambda_1 x}}{\Gamma(\alpha_1)}$ and $g_2(x) = \frac{\lambda_2^{\alpha_2} x^{\alpha_2 - 1} e^{-\lambda_2 x}}{\Gamma(\alpha_2)}$ in the proportion p_1 and $(1 - p_1)$ is given by $f(x) = \frac{p_1 \lambda_1^{\alpha_1} x^{\alpha_1 - 1} e^{-\lambda_1 x}}{\Gamma(\alpha_1)} + \frac{(1 - p_1) \lambda_2^{\alpha_2} x^{\alpha_2 - 1} e^{-\lambda_2 x}}{\Gamma(\alpha_2)}$.

Applying (1.2) $$\hat{p}_i^{new} = \frac{1}{n} \sum_{j=1}^{n} \frac{\hat{p}_i^{old} g_i(x_j)}{f(x_j)}$$

we get $$\hat{p}_1^{new} = \frac{1}{n} \sum_{j=1}^{n} \frac{\frac{\hat{p}_1 \lambda_1^{\alpha_1} x_j^{\alpha_1 - 1} e^{-\lambda_1 x_j}}{\Gamma(\alpha_1)}}{f(x_j)}$$

as the estimator for the first proportion parameter.

Although finding the estimate of $\hat{p}_1$ is straightforward, finding the estimates of $\alpha_1, \lambda_1, \alpha_2$ and λ_2 takes some effort. From (1.3), we know that the maximum likelihood estimators of in parameter must satisfy

$$\sum_{j=1}^{n} \frac{p_q \frac{\partial}{\partial \theta_{qm}} g_q(x_j)}{f(x_j)} = 0$$

Suppose we want to find an estimator for α_1, put $\theta_{qm} = \alpha_1$ and we require

$$\sum_{j=1}^{n} \frac{p_1 \frac{\partial}{\partial \alpha_1} \left(\frac{\lambda_1^{\alpha_1} x_j^{\alpha_1 - 1} e^{-\lambda_1 x_j}}{\Gamma(\alpha_1)} \right)}{f(x_j)} = 0$$

$$\sum_{j=1}^{n} \frac{p_1\left(\dfrac{e^{-\lambda_1 x_j}\Gamma(\alpha_1)[\lambda_1^{\alpha_1} x_j^{\alpha_1-1}\ln(x_j) + x_j^{\alpha_1-1}\lambda_1^{\alpha_1}.\ln\lambda_1] - \lambda_1^{\alpha_1} x_j^{\alpha_1-1} e^{-\lambda_1 x_j}\dfrac{d}{d\alpha_1}\Gamma(\alpha_1)}{(\Gamma(\alpha_1))^2}\right)}{f(x_j)} = 0$$

The differentiation of the gamma function in the equation above gives us some trouble. However, the problem can be solved by introducing the digamma function.

The digamma function $\psi(x)$ is defined as the logarithmic derivative of the gamma function, that is

$$\psi(x) = \frac{d}{dx}\log(\Gamma(x))$$

$$= \frac{1}{\Gamma(x)}\frac{d}{dx}\Gamma(x)$$

or $$\frac{d}{dx}\Gamma(x) = \psi(x)\Gamma(x)$$

After extracting the common factors, we can rewrite the equation above as

$$\sum_{j=1}^{n} \frac{p_1\left(\dfrac{e^{-\lambda_1 x_j}\lambda_1^{\alpha_1} x_j^{\alpha_1-1}[\Gamma(\alpha_1)\ln(\lambda_1 x_j) - \dfrac{d}{d\alpha_1}\Gamma(\alpha_1)]}{(\Gamma(\alpha_1))^2}\right)}{f(x_j)} = 0$$

$$\sum_{j=1}^{n} \frac{p_1\left(\dfrac{e^{-\lambda_1 x_j}\lambda_1^{\alpha_1} x_j^{\alpha_1-1}[\Gamma(\alpha_1)\ln(\lambda_1 x_j) - \psi(\alpha_1)\Gamma(\alpha_1)]}{(\Gamma(\alpha_1))^2}\right)}{f(x_j)} = 0$$

$$\sum_{j=1}^{n} \frac{e^{-\lambda_1 x_j}\lambda_1^{\alpha_1} x_j^{\alpha_1-1}[\ln(\lambda_1 x_j) - \psi(\alpha_1)]}{f(x_j)} = 0$$

$$\sum_{j=1}^{n} \frac{e^{-\lambda_1 x_j} \lambda_1^{\alpha_1} x_j^{\alpha_1-1} \ln(\lambda_1 x_j)}{f(x_j)} = \psi(\alpha_1) \sum_{j=1}^{n} \frac{e^{-\lambda_1 x_j} \lambda_1^{\alpha_1} x_j^{\alpha_1-1}}{f(x_j)}$$

$$\lambda_1^{\alpha_1} \sum_{j=1}^{n} \frac{e^{-\lambda_1 x_j} x_j^{\alpha_1-1} \ln(\lambda_1 x_j)}{f(x_j)} = \psi(\alpha_1) \sum_{j=1}^{n} \frac{e^{-\lambda_1 x_j} \lambda_1^{\alpha_1} x_j^{\alpha_1-1}}{f(x_j)}$$

$$\lambda_1^{\alpha_1} = \frac{\psi(\alpha_1) \sum_{j=1}^{n} \frac{e^{-\lambda_1 x_j} \lambda_1^{\alpha_1} x_j^{\alpha_1-1}}{f(x_j)}}{\sum_{j=1}^{n} \frac{e^{-\lambda_1 x_j} x_j^{\alpha_1-1} \ln(\lambda_1 x_j)}{f(x_j)}}$$

$$\alpha_1 \ln \lambda_1 = \ln \left(\frac{\psi(\alpha_1) \sum_{j=1}^{n} \frac{e^{-\lambda_1 x_j} \lambda_1^{\alpha_1} x_j^{\alpha_1-1}}{f(x_j)}}{\sum_{j=1}^{n} \frac{e^{-\lambda_1 x_j} x_j^{\alpha_1-1} \ln(\lambda_1 x_j)}{f(x_j)}} \right)$$

Finally, we get the maximum likelihood estimator for α_1,

$$\hat{\alpha}_1^{new} = \frac{\ln \left(\frac{\psi(\hat{\alpha}_1^{old}) \sum_{j=1}^{n} \frac{e^{-\lambda_1 x_j} \lambda_1^{\hat{\alpha}_1^{old}} x_j^{\hat{\alpha}_1^{old}-1}}{f(x_j)}}{\sum_{j=1}^{n} \frac{e^{-\lambda_1 x_j} x_j^{\hat{\alpha}_1^{old}-1} \ln(\lambda_1 x_j)}{f(x_j)}} \right)}{\ln \lambda_1}$$

Similarly, we have $\hat{\alpha}_2^{new} = \frac{\ln \left(\frac{\psi(\hat{\alpha}_2^{old}) \sum_{j=1}^{n} \frac{e^{-\lambda_2 x_j} \lambda_2^{\hat{\alpha}_2^{old}} x_j^{\hat{\alpha}_2^{old}-1}}{f(x_j)}}{\sum_{j=1}^{n} \frac{e^{-\lambda_2 x_j} x_j^{\hat{\alpha}_2^{old}-1} \ln(\lambda_2 x_j)}{f(x_j)}} \right)}{\ln \lambda_2}$

In order to estimate λ_1, we proceed by setting

$$\sum_{j=1}^{n} \frac{p_1 \frac{\partial}{\partial \lambda_1}\left(\frac{\lambda_1^{\alpha_1} x_j^{\alpha_1-1} e^{-\lambda_1 x_j}}{\Gamma(\alpha_1)}\right)}{f(x_j)} = 0$$

$$\sum_{j=1}^{n} \frac{\frac{p_1 x_j^{\alpha_1-1}}{\Gamma(\alpha_1)}\left(\alpha_1 \lambda_1^{\alpha_1-1} e^{-\lambda_1 x_j} - \lambda_1^{\alpha_1} e^{-\lambda_1 x_j} x_j\right)}{f(x_j)} = 0$$

$$\sum_{j=1}^{n} \frac{x_j^{\alpha_1-1}\left(\alpha_1 e^{-\lambda_1 x_j} - \lambda_1 e^{-\lambda_1 x_j} x_j\right)}{f(x_j)} = 0$$

$$\sum_{j=1}^{n} \frac{x_j^{\alpha_1-1}\left(\alpha_1 e^{-\lambda_1 x_j}\right)}{f(x_j)} = \lambda_1 \sum_{j=1}^{n} \frac{x_j^{\alpha_1-1} e^{-\lambda_1 x_j} x_j}{f(x_j)}$$

$$\hat{\lambda}_1^{new} = \frac{\sum_{j=1}^{n} \frac{x_j^{\alpha_1-1}\left(\alpha_1 e^{-\hat{\lambda}_1^{old} x_j}\right)}{f(x_j)}}{\sum_{j=1}^{n} \frac{x_j^{\alpha_1} e^{-\hat{\lambda}_1^{old} x_j}}{f(x_j)}}$$

Similarly, we get

$$\hat{\lambda}_2^{new} = \frac{\sum_{j=1}^{n} \frac{x_j^{\alpha_2-1}\left(\alpha_2 e^{-\hat{\lambda}_2^{old} x_j}\right)}{f(x_j)}}{\sum_{j=1}^{n} \frac{x_j^{\alpha_2} e^{-\hat{\lambda}_2^{old} x_j}}{f(x_j)}}$$

The mean and variance of a gamma distribution with probability density function $g_1(x) = \frac{\lambda_1^{\alpha_1} x^{\alpha_1 - 1} e^{-\lambda_1 x}}{\Gamma(\alpha_1)}$ are given by $\frac{\alpha_1}{\lambda_1}$ and $\frac{\alpha_1}{\lambda_1^2}$ respectively. We will first calculate the sample means and sample variances for both the intermetallic region and gold region. An estimate of α_1 is then given by $\frac{(\text{sample mean of the intermetallic region})^2}{\text{variance of the intermetallic region}}$ and the estimate for λ_1 is $\frac{\text{sample mean of the intermetallic region}}{\text{variance of the intermetallic region}}$. Likewise, the estimates of α_2 and λ_2 are given by $\frac{(\text{sample mean of the gold region})^2}{\text{variance of the gold region}}$ and $\frac{\text{sample mean of the gold region}}{\text{variance of the gold region}}$ respectively.

Appendix D

Output for the analysis on the reduced images

Image	Initial estimates of p	mu1	sigma1	mu2	sigma2	Final estimates of p	mu1	sigma1	mu2	sigma2
S101a	0.55136	131	484.55	207	253.8	0.49887	137.42	318.67	205.88	363.79
S102a	0.66941	123	502.89	201	228.91	0.56397	127.62	188.93	195.64	445.84
S103a	0.64415	136	369.73	213	243.1	0.60565	137.79	307.38	205.46	332.9
S104a	0.58395	134	491.22	222	282.57	0.63629	149.13	495.27	220.16	172.33
S105a	0.64639	132	475.54	202	229.46	0.55894	136.3	299.25	198.66	395.63
S106a	0.59927	144	366.64	221	322.59	0.65117	148.83	496.54	218.78	224.8
S201a	0.73762	168	750.42	182	1104.3	0.87911	156.88	641.05	225.99	117.77
S202a	0.79147	144	232.91	236	635.17	0.7994	144.48	251.98	219.93	325.46
S203a	0.81252	157	340.08	244	1106.6	0.85436	151.54	352.1	224.59	218.29
S204a	0.82714	161	349.12	182	1567.7	0.88196	155.17	355.54	227.21	177.39
S205a	0.84881	154	396.94	223	410.57	0.87508	146.03	361.26	219.83	248.53
S206a	0.86707	142	265.24	211	338.46	0.84657	138.4	238.52	206.87	515.05
S301a	0.6597	118	602.48	220	271.11	0.60672	128.66	288.75	209.45	450.67
S302a	0.68518	115	540.4	212	192.78	0.59703	122.3	166.91	198.96	465.28
S303a	0.65032	117	543.32	212	194.66	0.59189	125.73	233.49	203.69	353.95
S304a	0.64014	108	687.16	205	186.93	0.53836	116	136.8	197.09	486.38
S305a	0.68648	111	545.29	208	175.94	0.58445	116.17	129.17	195.14	519.2
S306a	0.66325	120	522.32	219	256.51	0.60682	128.04	262.03	207.88	439.91
S401a	0.69259	111	490.77	211	240.75	0.5891	116.58	98.175	199.39	761.99
S402a	0.70644	112	525.26	215	231.6	0.5904	118.1	105.43	196.31	735.06
S403a	0.67551	113	522.8	213	218.78	0.57019	117.95	123.23	199.35	657.7
S404a	0.69585	109	588.77	213	211.66	0.61034	118.14	146.27	200.2	558.28
S405a	0.66926	107	617.67	223	297.35	0.58922	116.81	147.22	205.94	655.06
S406a	0.69542	111	498.54	215	229.2	0.57571	114.49	83.981	196.41	800.07
S501a	0.48699	167	523.06	213	308.77	0.59374	161.61	586.06	211.8	314.34
S502a	0.47691	178	908.37	186	604.01	0.56803	162.91	640.72	205.21	303.16
S503a	0.50779	181	981.23	185	595.24	0.6728	167.16	677.4	205.18	301.3
S504a	0.4967	164	478.35	210	265.12	0.62815	162.81	649.6	210.88	272.95
S505a	0.62182	160	422.47	182	787.74	0.62857	154.44	488.56	201.74	409.95
S506a	0.58309	161	561.27	215	312.37	0.63478	152.41	555.28	210.41	305.08

Table D1 Initial and final estimates of parameters in the normal mixture model for the reduced images.

Image	Bonder	Normal Mixture Threshold	%IM	Gamma Mixture Threshold	%IM
S101a	A	171	0.50294	185	0.5690
S102a	A	158	0.57415	176	0.6446
S103a	A	174	0.61317	186	0.6635
S104a	A	194	0.63186	194	0.6319
S105a	A	168	0.57053	185	0.6666
S106a	A	192	0.64998	194	0.6587
S201a	B	211	0.87488		
S202a	B	186	0.80202	193	0.8175
S203a	B	199	0.85525	205	0.8684
S204a	B	204	0.88193	209	0.8920
S205a	B	194	0.87673	204	0.8962
S206a	B	177	0.85593	193	0.8924
S301a	C	168	0.6118	188	0.6814
S302a	C	155	0.60408	181	0.6841
S303a	C	163	0.59465	182	0.6524
S304a	C	147	0.54577	177	0.6281
S305a	C	146	0.59183	177	0.6757
S306a	C	166	0.61129	187	0.6815
S401a	D	143	0.5963	178	0.6839
S402a	D	145	0.60188	179	0.6994
S403a	D	147	0.57846	180	0.6712
S404a	D	150	0.61588	179	0.6895
S405a	D	150	0.59663	183	0.6752
S406a	D	139	0.58309	178	0.6855
S501a	E	192	0.58959	190	0.5704
S502a	E	187	0.54194	186	0.5299
S503a	E	194	0.66005	187	0.5756
S504a	E	193	0.60865	189	0.5695
S505a	E	184	0.64719	194	0.7385
S506a	E	188	0.63899	192	0.6694

Table D2 Comparison of threshold points and intermetallic percentage for both normal mixture and gamma mixture model in the reduced image.

Image	Initial estimates of					Final estimates of				
	p	alpha1	lamda1	alpha2	lamda2	p	alpha1	lamda1	alpha2	lamda2
S101a	0.5514	5.322	0.1166	53.850	0.4686	0.5988	3.7258	0.0734	53.776	0.4657
S102a	0.6694	3.924	0.1010	55.864	0.5093	0.6572	3.8942	0.1012	55.852	0.5180
S103a	0.6442	5.654	0.1262	57.108	0.5004	0.6871	4.1438	0.0852	57.091	0.4958
S104a	0.5840	7.010	0.1392	58.508	0.4816	0.6568	4.6615	0.0825	59.049	0.4730
S105a	0.6464	5.270	0.1151	57.242	0.5164	0.7078	3.5913	0.0697	57.153	0.5163
S106a	0.5993	6.889	0.1373	49.986	0.4158	0.6827	4.4926	0.0788	50.397	0.4069
S201a	0.7376	7.215	0.1320	34.663	0.3041	0.9279	4.1557	0.0628	35.166	0.2857
S202a	0.7915	10.315	0.2104	45.007	0.3634	0.8181	8.2562	0.1632	45.368	0.3554
S203a	0.8125	10.461	0.1919	39.221	0.3214	0.8763	6.8739	0.1181	40.640	0.3111
S204a	0.8271	11.863	0.2054	34.826	0.2881	0.8998	7.8653	0.1277	36.848	0.2790
S205a	0.8488	7.961	0.1603	42.028	0.3513	0.9066	5.2188	0.0979	42.894	0.3359
S206a	0.8671	7.579	0.1717	41.190	0.3502	0.8941	5.6776	0.1239	41.404	0.3393
S301a	0.6597	3.402	0.0917	58.008	0.4818	0.6945	2.7305	0.0675	58.062	0.4751
S302a	0.6852	2.906	0.0882	71.890	0.6352	0.6886	2.8855	0.0861	71.874	0.6374
S303a	0.6503	3.168	0.0912	69.762	0.6106	0.6645	2.7392	0.0753	69.739	0.6099
S304a	0.6401	1.845	0.0645	67.293	0.6036	0.6326	1.9048	0.0671	67.259	0.6105
S305a	0.6865	1.958	0.0699	71.725	0.6471	0.6794	2.0945	0.0755	71.680	0.6540
S306a	0.6633	3.480	0.0949	60.984	0.5120	0.6915	2.8886	0.0733	61.023	0.5063
S401a	0.6926	2.259	0.0807	58.452	0.4961	0.6841	2.7318	0.0996	58.377	0.5005
S402a	0.7064	2.505	0.0837	62.743	0.5421	0.7013	2.8420	0.0957	62.673	0.5459
S403a	0.6755	2.316	0.0777	62.679	0.5380	0.6756	2.4776	0.0824	62.643	0.5401
S404a	0.6959	2.234	0.0776	64.998	0.5633	0.6948	2.3831	0.0824	64.965	0.5660
S405a	0.6693	1.986	0.0724	58.683	0.4845	0.6770	2.0133	0.0709	58.683	0.4835
S406a	0.6954	1.926	0.0714	62.547	0.5368	0.6872	2.3557	0.0891	62.481	0.5417
S501a	0.4870	10.002	0.1707	44.406	0.3896	0.6321	5.6592	0.0802	44.169	0.3831
S502a	0.4769	9.733	0.1630	52.620	0.4770	0.6016	5.7087	0.0805	52.274	0.4786
S503a	0.5078	10.667	0.1754	52.035	0.4768	0.6504	6.2825	0.0869	51.682	0.4812
S504a	0.4967	9.330	0.1599	48.884	0.4339	0.6388	5.2522	0.0748	48.710	0.4307
S505a	0.6218	9.067	0.1592	46.490	0.4228	0.7825	5.2341	0.0770	46.157	0.4213
S506a	0.5831	7.105	0.1333	47.038	0.4127	0.7092	4.2725	0.0675	47.015	0.4060

Table D3 Initial and final estimates of parameters in the gamma mixture model for the reduced images.

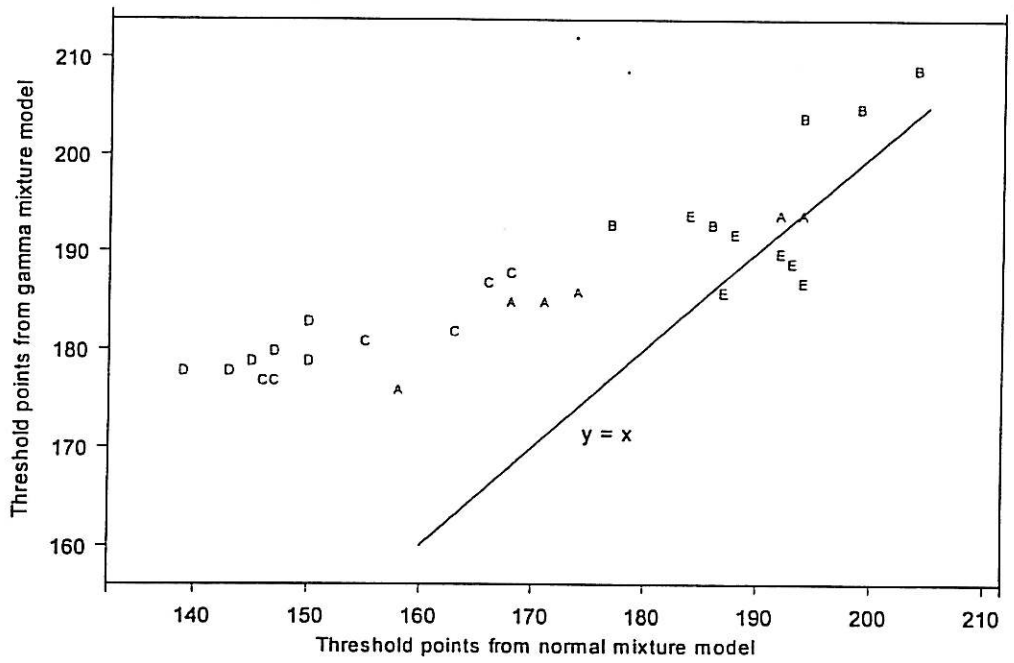


Figure D1 Scatter plot of threshold points from the reduced images.

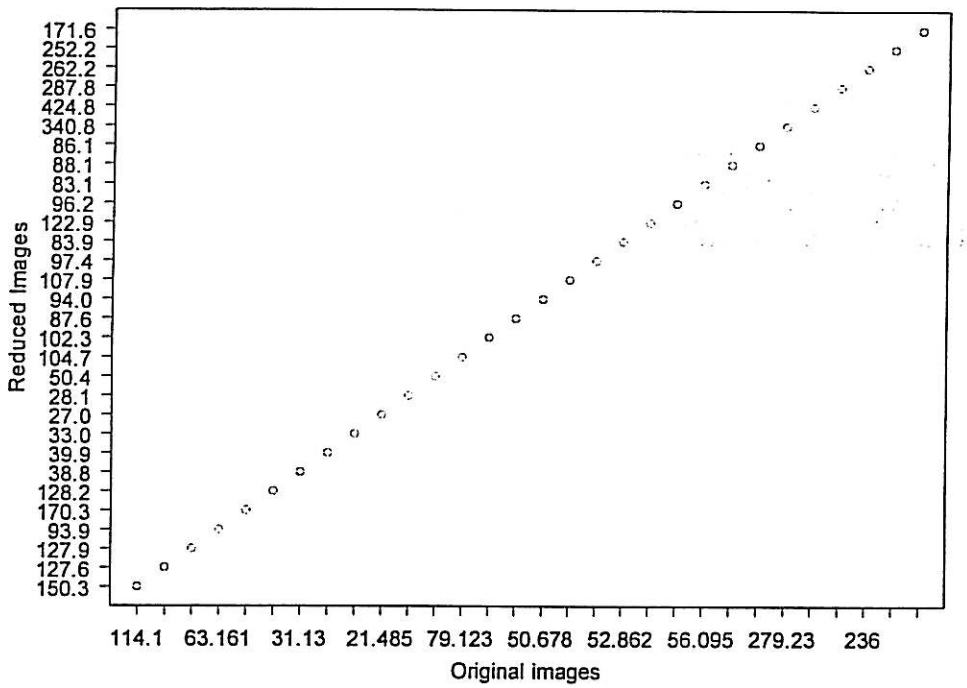


Figure D2 Scatter plot of compactness from both original images and the reduced images.

Appendix E

The Matlab Programme (1) —Fitting normal mixture model

```
% This programme is used to fit a two component normal mixture model to the
   brightness histogram and use minimum expected cost method and minimum error
   method in finding the threshold points and measures of interest.

for q=1:30    %The calculations loop for 30 times to handle all 30 images

image=input('Input the complete filename of the image :','s'); % Read in the image
image=imread(image);
imshow(image);

t1=clock;     % Capture the starting time

% Transforming the image into grey-scale image
A=(double(image(:,:,1))*0.299+double(image(:,:,2))*0.587+
    double(image(:,:,3))* 0.114)/255;

% Produce the brightness histogram for the grey scale image.
image_w_background=imhist(A);
figure, bar(image_w_background);

% Capturing brightness levels from 96/255 to 249/255
real_image=image_w_background(97:250,:);
figure,bar(real_image);

% Convert histogram to probability density
S=sum(real_image);
image_density=real_image/S;

% Preset initial threshold point corresponding to brightness level 180/255.
threshold=86;

% The brightness from 0 to the threshold point representing the intermetallic phase
B=[ones(1,threshold) zeros(1,154-threshold)];
intermetallic_region=B.*real_image';
intermetallic_density=B.*image_density';

% The brightness larger than the threshold point representing the gold metal
C=[zeros(1,threshold) ones(1,154-threshold)];
gold_region=C.*real_image';
gold_density=C.*image_density';
```

```
% Initial estimate of p by the percentage of intermetallic phase
p=sum(intermetallic_density);

% Initial estimate of mu1 and mu2 by the most frequently appear brightness in the
   intermetallic region and gold region.

mu1=min(find(intermetallic_region==max(intermetallic_region)));
mu2=min(find(gold_region==max(gold_region)));

D=[1:154];

% Estimate the variances by the unbiased estimators
sigma1=sum(((D-mu1).*B).^2.*intermetallic_region)/(sum(intermetallic_region)-1);
sigma2=sum((((D-mu2).*C).^2).*gold_region)/(sum(gold_region)-1);

% mu + 95 gives the corresponding original position of the means
i_e=[p mu1+95 sigma1 mu2+95 sigma2];

% Indicator variables
p_new=1;
p_old=0;

% z is the number of iterations
z=0;

% The EM Algorithm
while ((abs(p_new - p_old) > 0.00001) || (abs(mu1_new - mu1_old) > 0.0001) ||
(abs(sigma1_new-sigma1_old) > 0.001) || (abs(mu2_new-mu2_old)>0.0001)) ||
(abs(sigma2_new-sigma2_old) > 0.001)
        p_old=p;
        mu1_old=mu1;
        mu2_old=mu2;
        sigma1_old=sigma1;
        sigma2_old=sigma2;
        v1=1/(2*pi*sigma1)^0.5;
        v2=1/(2*pi*sigma2)^0.5;
        pdf=p*v1*exp(-0.5/sigma1*(D-mu1).^2)+(1-p)*v2*exp(-0.5/sigma2*
            (D-mu2).^2);
        p=sum(p*v1*exp(-0.5/sigma1*(D-mu1).^2).*real_image'./pdf)/S;
        mu1=sum((exp(-0.5/sigma1*(D-mu1).^2).*D.*real_image')./pdf)/
              sum((exp(-0.5/sigma1*(D-mu1).^2).*real_image')./pdf);
        mu2=sum((exp(-0.5/sigma2*(D-mu2).^2).*D.*real_image')./pdf)/
              sum((exp(-0.5/sigma2*(D-mu2).^2).*real_image')./pdf);
        sigma1=sum((p*exp(-0.5/sigma1*(D-mu1).^2).*(D-mu1).^2.*real_image')
                  ./pdf)/sum((p*exp(-0.5/sigma1*(D-mu1).^2).*real_image')./pdf);
        sigma2=sum(((1-p)*exp(-0.5/sigma2*(D-mu2).^2).*(D-mu2).^2
                  .*real_image')./pdf)/sum(((1-p)*exp(-0.5/sigma2*(D-mu2).^2).*
                  real_image')./pdf);
```

```
        p_new=p;
        mu1_new = mu1;
        sigma1_new=sigma1;
        mu2_new = mu2;
        sigma2_new=sigma2;
        z=z+1;
end;

% Final estimates of parameters
f_e=[p mu1+95 sigma1 mu2+95 sigma2];

% Output of graphs showing component pdf and cdf, KS statistic is calculated.
v1=1/(2*pi*sigma1)^0.5;
v2=1/(2*pi*sigma2)^0.5;
pdf1=p*v1*exp(-0.5/sigma1*(D-mu1).^2);
pdf2=(1-p)*v2*exp(-0.5/sigma2*(D-mu2).^2);

E=[image_density';pdf];
figure, plot(D,E);

image_cdf=cumsum(image_density');
cdf=p*normcdf(D,mu1,sqrt(sigma1))+(1-p)*normcdf(D,mu2,sqrt(sigma2));

F=[image_cdf; cdf];
figure, plot(D,F);

ks=max(abs(image_cdf-cdf));
acceptance_limit=1.63/sqrt(S);

% Finding the threshold point using minimum expected cost method
pdf1=v1*exp(-0.5/sigma1*(D-mu1).^2);
pdf2=v2*exp(-0.5/sigma2*(D-mu2).^2);
ratio=pdf1./pdf2;
criteria=(1-p)/p;

% Searching the threshold point from brightness level 135/255 onwards
i=40;
threshold=0;
while threshold==0
     i=i+1;
    if ratio(i)<criteria
        threshold=i+95;
    end;
end;
```

```
% Generating binary image by using threshold value obtained above
for i=1:576
    for j=1:768
        if (A(i,j)<threshold/255)
            I(i,j)=0;
        else
            I(i,j)=1;
        end;
    end;
end;

% Edge detection algorithm
for i=1:575
    for j=1:767
        if abs(2*I(i,j)-I(i+1,j)-I(i,j+1))>0
            boundary(i,j)=0;
        else
            boundary(i,j)=1;
        end;
    end;
end;

% Finding 'perimeter' and 'area'
edge=sum(abs(boundary-1));
edge=sum(edge);
intermetallic=sum(abs(I-1));
intermetallic=sum(intermetallic);
ipercent=intermetallic/S;
compactness=edge^2/(12.566*intermetallic);

% Printout of results and binary images
figure, imshow(I);
figure, imshow(boundary);


% Minimum error thresholding
for i=20:120;
    histogram1=image_density(1:i);
    histogram2=image_density(i+1:154);
    weight1=[1:i];
    weight2=[i+1:154];
    p=sum(histogram1);
    mu1=sum(histogram1'.*weight1)/p;
    mu2=sum(histogram2'.*weight2)/(1-p);
    sigma1=(sum((weight1-mu1).^2.*histogram1')/p);
    sigma2=(sum((weight2-mu2).^2.*histogram2')/(1-p));
    T(i-19)=p*log(sigma1)+(1-p)*log(sigma2)-p*log(p)-(1-p)*log(1-p);
end;
```

```
threshold1=find(T==min(T))+114;

% Generating binary image by using threshold value above
for i=1:576
    for j=1:768
        if (A(i,j)<threshold1/255)
           I1(i,j)=0;
        else
           I1(i,j)=1;
        end;
    end;
end;

% Edge detection algorithm
for i=1:575
    for j=1:767
        if abs(2*I1(i,j)-I1(i+1,j)-I1(i,j+1))>0
           boundary1(i,j)=0;
        else
           boundary1(i,j)=1;
        end;
    end;
end;

% Finding 'perimeter' and 'area'
edge=sum(abs(boundary1-1));
edge=sum(edge);
intermetallic=sum(abs(I1-1));
intermetallic=sum(intermetallic);
ipercent1=intermetallic/S;
compactness1=edge^2/(12.566*intermetallic);
timeused=etime(clock,t1);

% Printout of results and binary images
M(q,:)=[i_e f_e ks acceptance_limit threshold ipercent compactness threshold1
ipercent1 compactness1 timeused z];
figure, imshow(I1);
figure, imshow(boundary1);
end;

% M is a matrix that stores all the numerical output
M
```

Appendix F

The Matlab Programme (2) —Fitting gamma mixture model

```
% This programme is used to fit a two component gamma mixture model to the
   brightness histogram and use minimum expected cost method in finding the
   measures of interest.

format long

% Reading the image
image=input('Image:','s');
image=imread(image);

% Transforming the image into grey-scale image
A=(double(image(:,:,1))*0.299+double(image(:,:,2))*0.587+
    double(image(:,:,3))*0.114)/255;

%Capturing the starting time
t1=clock;
image_w_background=imhist(A);

% Capturing the part of histogram containing needed information
real_image=image_w_background(97:250,:);
real_image=real_image';

% Converting histogram to probability density
S=sum(real_image);
image_density=real_image/S;

threshold=86;

% The brightness from 0 to the threshold point representing the intermetallic phase
B=[ones(1,threshold) zeros(1,154-threshold)];
intermetallic_region=B.*real_image;
intermetallic_density=B.*image_density;

% The brightness larger than the threshold point representing gold phase
C=[zeros(1,threshold) ones(1,154-threshold)];
gold_region=C.*real_image;
gold_density=C.*image_density;

image_cdf=cumsum(image_density');

D=[1:154];
```

```
% Estimate the variances by the unbiased estimators
p=sum(intermetallic_density);
mean1=sum(D.*intermetallic_region)/sum(intermetallic_region);
mean2=sum(D.*gold_region)/sum(gold_region);
sigma1=sum((((D-mean1).*B).^2).*intermetallic_region)/
        (sum(intermetallic_region)-1);
sigma2=sum((((D-mean2).*C).^2).*gold_region)/(sum(gold_region)-1);

% Initial estimates of parameters
alpha1=mean1^2/sigma1;
alpha2=(mean2)^2/sigma2;
lamda1=mean1/sigma1;
lamda2=mean2/sigma2;
p_new=1;
p_old=0;
i_ne=[p alpha1 lamda1 alpha2 lamda2];

% z is the number of iterations
z=1;

% The EM Algorithm
while (abs(p_new - p_old) > 0.00001) || (abs(lamda1_new-lamda1_old) > 0.005) ||
(abs(lamda2_new-lamda2_old) > 0.005) || (abs(alpha1_new - alpha1_old) > 0.005) ||
(abs(alpha2_new-alpha2_old)>0.005)
    p_old=p;
    alpha1_old=alpha1;
    alpha2_old=alpha2;
    lamda1_old=lamda1;
    lamda2_old=lamda2;
    v1=lamda1^alpha1/gamma(alpha1);
    v2=lamda2^alpha2/gamma(alpha2);
    pdf=p*v1*exp(-lamda1*D).*(D.^(alpha1-1))+(1-p)*v2*
        exp(-lamda2*D).*(D.^(alpha2-1));
    p=sum(exp(-lamda1*D).*(D.^(alpha1-1)).*real_image./pdf)*p*v1/S;
    lamda1=sum((alpha1*(D.^(alpha1-1)).*exp(-lamda1*D))./pdf.*real_image)/
            sum(((D.^alpha1).*exp(-lamda1*D))./pdf.*real_image);
    lamda2=sum((alpha2*(D.^(alpha2-1)).*exp(-lamda2*D))./pdf.*real_image)/
            sum(((D.^alpha2).*exp(-lamda2*D))./pdf.*real_image);
    alpha1=log((psi(alpha1)*sum((gampdf(D,alpha1,1/lamda1)*p)./
            pdf.*real_image))/sum((p*(exp(-lamda1*D).*(D.^(alpha1-1)).*
            log(lamda1*D)./gamma(alpha1))./pdf.*real_image)))/log(lamda1);
    alpha2=log((psi(alpha2)*sum((gampdf(D,alpha2,1/lamda2)*(1-p))./
            pdf.*real_image))/sum(((1-p)*(exp(-lamda2*D).*(D.^(alpha2-1)).*
            log(lamda2*D)./gamma(alpha2))./pdf.*real_image)))/log(lamda2);

    p_new=p;
    lamda1_new=lamda1;
    lamda2_new=lamda2;
```

```
        alpha1_new=alpha1;
        alpha2_new=alpha2;
        z=z+1;
end;

f_ne=[p alpha1 lamda1 alpha2 lamda2];

G==[1:154];

% Output of graphs showing component pdf and cdf, KS statistic is calculated.
pdf=p*v1*exp(-lamda1*G).*(G.^(alpha1-1))+(1-p)*v2*exp(-lamda2*G).*
    (G.^(alpha2-1));
pdf1=p*v1*exp(-lamda1*G).*(G.^(alpha1-1));
pdf2=(1-p)*v2*exp(-lamda2*G).*(G.^(alpha2-1));
image_density1=real_image/S;
H=[image_density1;pdf];
figure, plot(D,H);

cdf=p*gamcdf(G,alpha1,1/lamda1)+(1-p)*(gamcdf(G,alpha2,1/lamda2));
image_cdf=image_cdf';
F=[image_cdf; cdf];
figure, plot(D,F);

% Finding the Kolmogolov-Smirnov Statistic
ks=max(abs(image_cdf-cdf));
acceptance_limit=1.63/sqrt(S);

v1=lamda1^alpha1/gamma(alpha1);
v2=lamda2^alpha2/gamma(alpha2);
pdf1=v1*exp(-lamda1*D).*(D.^(alpha1-1));
pdf2=v2*exp(-lamda2*D).*(D.^(alpha2-1));

% Minimum expected cost threholding
ratio=pdf1./pdf2;
criteria=(1-p)/p;

i=30;
threshold=0;

while threshold==0
  i=i+1;
  if ratio(i)<criteria
     threshold=i+95;
  end;
end;
```

```
% Generating binary image by using threshold value above
for i=1:576
    for j=1:768
        if (A(i,j)<threshold/255)
           I(i,j)=0;
        else
           I(i,j)=1;
        end;
    end;
end;


% Edge detection algorithm
for i=1:575
    for j=1:767
        if abs(2*I(i,j)-I(i+1,j)-I(i,j+1))>0
           boundary(i,j)=0;
        else
           boundary(i,j)=1;
        end;
    end;
end;


% Finding 'perimeter' and 'area'
edge=sum(abs(boundary-1));
edge=sum(edge);
intermetallic=sum(abs(I-1));
intermetallic=sum(intermetallic);
ipercent=intermetallic/S;
compactness=edge^2/(12.566*intermetallic);
compactness1=1/compactness;

timeused=etime(clock,t1);

% J is the matrix containing all numerical output
J=[i_ne f_ne ks acceptance_limit threshold ipercent compactness1 timeused z]
figure, imshow(I);
figure, imshow(boundary);
end;
```